A NOVEL FRAMEWORK FOR UNDERSTANDING ATYPICAL IMAGES

BY BABAK SALEH

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

Written under the direction of

Ahmed Elgammal

and approved by

_____

_____

_____

_____

New Brunswick, New Jersey

January, 2017

ABSTRACT OF THE DISSERTATION

A Novel Framework for Understanding Atypical Images

by BABAK SALEH

Dissertation Director: Ahmed Elgammal

In the past few years, there has been a tremendous amount of progress in the field of computer vision. As of now, we have reliable object detectors and classifiers that can recognize thousands of object categories. However, the ultimate goal of computer vision is to build systems that can understand and reason about images, far beyond scene categorization and object detection. In this thesis, algorithms have been proposed to empower computers with the human-level ability of detecting and reasoning about images that are understudied in the mainstream computer vision community.

In chapter 1, we open the conversation about abnormality detection, by discussing how humans form visual concepts (e.g. an object category) and perceive meaningful deviations from these learned concepts as signals for abnormality. However, there is not a comprehensive study about what factors lead humans in this decision-making process. In chapter 2 we collect the first dataset of abnormal images from the web. Conduct several human subject experiments, and perform a thorough set of analysis to discover hidden factors in human judgment about abnormality. These analyses lead us to propose a taxonomy of comprehensive reasons of abnormality in images.

Inspired by human reasoning, we address the problem of detecting abnormal objects and reasoning about their abnormality in terms of visual attributes, such as irregular shape, texture

or color (chapter 3). Although our computational models are learned without seeing any abnormal objects at training time, but still are capable of detecting and reasoning about abnormal images at the test time. In chapter 4 we develop probabilistic frameworks to model typical images and find atypical images as a meaningful deviation from this model. In chapter 5, we use the typicality scores of images and objects to improve the generalization capacity of the state- of-the-art Convolutional Neural Networks (CNN) for the task of object classification. We train these CNN models by minimizing a weighted loss function that incorporates in the typicality scores of samples. Our experiments show that this training strategy results in more generalized classifiers, which can be applied even to the extent of abnormal images.

In chapter 6 of this thesis, we study two problems that extend our framework for abnormality detection to special cases. We develop algorithms for detecting and localizing attributes in images. In addition to the application of localized attributes for the problem of abnormality detection, we show that fine-grained object categorization benefit from such rich information as well. We also propose algorithms to learn visual classifiers directly from the textual description of an object category. This zero-shot learning strategy extends the abnormality detection framework to object categories that are not present at the time of training. We close this thesis by discussing the main contributions and some future work.

Acknowledgements

This dissertation would not have been possible without the support of many people. First of all, Professor Ahmed Elgammal who gave me the opportunity of working on interesting problems, taught me conducting scientific research, enriched my learning experiences with a perfect balance of freedom and supervision, and helped me through the challenges of my graduate studies. I also want to thank my thesis committee members, Professor Jacob Feldman who always provided me with fascinating insights and thoughtful comments. Professor Kostas Bekris and Professor Devi Parikh who not only oversaw my thesis, but also showed me outstanding examples of academic scholars. I would like to express my gratitude to Professor Mehrdad Shahshahani who has been my role model since my undergraduate studies; and Professor Ali Farhadi who I have learned substantially from his mentorship and collaboration.

I also thank my precious friends and colleagues who made my time at Rutgres, a memorable experience: Shahriar, Afra, Tarek, Hooman, Mohsen, Reza, Brian, Kana, Amir, Soheil, Nader, Behnam, Mohamed, Turgay, Ali, Chetan, and Lin. Special thanks goes to Negar for her constant support, which helped me to navigate through difficult times. I would not be where I am today without the amazing support, encouragement and unconditional love from my parents, Mojhgan and Shahpour. You gave me the confidence and motivation to follow my dreams and passion to build a bright future. Lastly, I am very lucky to have an extremely talented brother, Behrad, who always keeps me motivated, excited, and happy.

Dedication

To my family.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Human Visual Understanding and Prototype Theory

Humans begin to form categories and abstractions at an early age [108]. The mechanisms underlying human category formation are the subject of many competing accounts, including those based on prototypes [106], exemplars [115], density estimation [6], and Bayesian inference [68]. But all modern models agree that human category representations involve subjective variations in the typicality or probability of objects within categories. For example, bird category includes both highly typical examples such as robins, as well as extremely atypical examples like penguins and ostriches, which while belonging to the category seem like subjectively "abnormal" examples. Visual images can seem abnormal, in that they can exhibit features that depart in some way from what is typical for the categories to which they belong. In this thesis, we ask what makes visual images seem abnormal with respect to their apparent categories, something that human observers can readily judge but is difficult to capture computationally.

The way humans form a visual concept in their mind is still not well defined. However, researchers have shown that this learning process varies from one category to another category, based on how entry-level an object category is. For example, humans learn the concept of cars by generalization from different samples of cars. But they learn the concept of sedan cars by discriminating its samples from SUVs. For the first case, cars can be defined as a 3D boxy shape objects made by steel and glasses. While for the later case, SUVs have a higher roof rather than sedans and usually are bigger. These different types of learning refer to intra vs. inter class variability.

The categories of objects might show significant within-category diversity. The typicality [1]

[1] We will use typicality/atypicality when referring to objects, scenes and context, while we will use normality/abnormality when referring to images. However, at some points we use these words interchangeably.

is a graded phenomenon, in which objects can be extremely typical (close to prototype), moderately typical (fairly close), atypical (not close), or borderline category members (objects that are about equally distant from two different prototypes). Some members of a category might be considered as more prototypical examples. For example, humans affirm robin as a prototype of birds more than chickens, even though chickens are more frequently seen than robins. This fact refers to the notion of typicality of an object for a given concept. Generally speaking, typical category members are the common examples- what a person would normally think of when he or she thinks of the category- and atypical objects are ones that are known to be members but are uncommon in some way. In the above example, birds typically can fly, while chicken cannot.

A diverse set of reasons may cause abnormality. An object can be abnormal due to the absence of typical attributes (a car without wheels) or the presence of atypical attributes (a car with wings). Also, abnormality can be caused by deviations from the extent by which an attribute varies inside a category (a furry dog). Furthermore, contextual irregularities and semantical peculiarities can also cause abnormalities such as an elephant in the room [164, 161]. On the one hand, in chapter 3 we mainly focus on abnormalities stemming from the object itself, not from the context around the object. On the other hand, in chapter 4 we propose computational models to quantify atypicalities in images.

We derive our computational model for finding and reasoning about abnormality in images without using abnormal images. Contrary to the traditional approach to abnormality (outlier) detection in the field of machine learning (e.g. fraud detection), we cannot train our model with abnormal samples. However we are able to detect abnormal images during test time using our model. This assumption is rooted in the fact that images are more diverse and has high dimensionality rather than other types of data (e.g. financial data). As a result it is impractical to capture all aspects of abnormality in images and we cannot build a comprehensive model for atypicality. This learning principle is in line with human perception as well. Humans can easily spot an atypical sofa even when they observe it for the first time. Due to aforementioned reasons; we build a computational model to measure how normal an image looks like. This measurement would eventually detect abnormal images as samples with low score of normality.

We form the aforementioned computational model for normal objects (chapter 3) and images (chapter 4) by finding the most common and crucial visual attributes for a given object class. Visual attributes have been used for a variety of tasks in the field of computer vision; such as object categorization, zero shot learning, event detection in videos, etc. The most common usage of visual attributes is object description. For example, a car can be described as a 3D boxy shape object, which has wheels, side handlebar, mirrors and its body is made by metal and glass. Each of these attributes can be visualized by showing various examples from different categories. For example, we can learn how a wheel look like by looking at cars, bikes, motorbikes, airplanes, etc. This helps us to learn a visual model for each of them independent of object categories. Later we can describe a new category via these learned visual classifiers. We used this advantage to model abnormal objects, even though we have not seen them before. Additionally we extend the definition of visual attributes to include the location information of the detected attribute as well, where we show "Localized Attributes" are more powerful tools for object recognition (chapter 6).

What does studying abnormality in images tell us about object recognition? Despite the superior performance of the state-of-the-art in object detection and recognition, but these algorithms are missing the important ability of humans for reasoning about their decision, especially for the case of atypical objects [134, 85]. Humans seem to be able to recognize abnormalities and reason about category memberships of atypical instances without learning on any atypical instance [103]. Can state-of-the-art computer vision object categorization and detection algorithms generalize as well to atypical images? In contrast to humans ability to generalize and successfully categorize atypical instances, state-of-the-art computer vision algorithms fail to achieve similar generalization. Table 5.1 shows categorization results of several state-of-the-art approaches [89, 148, 80, 152] when tested on our dataset of abnormal images. In chapter 5 we argue that studying generalization to atypical images, without optimizing on them, provides insights on how a recognition algorithm might simulate human performance.

Abnormality detection plays a substantial role in broad range of tasks: learning visual concepts [146], natural languages processing [72], human perception and cognition [17], human action recognition [102], etc. In addition, there are various applications for developing an

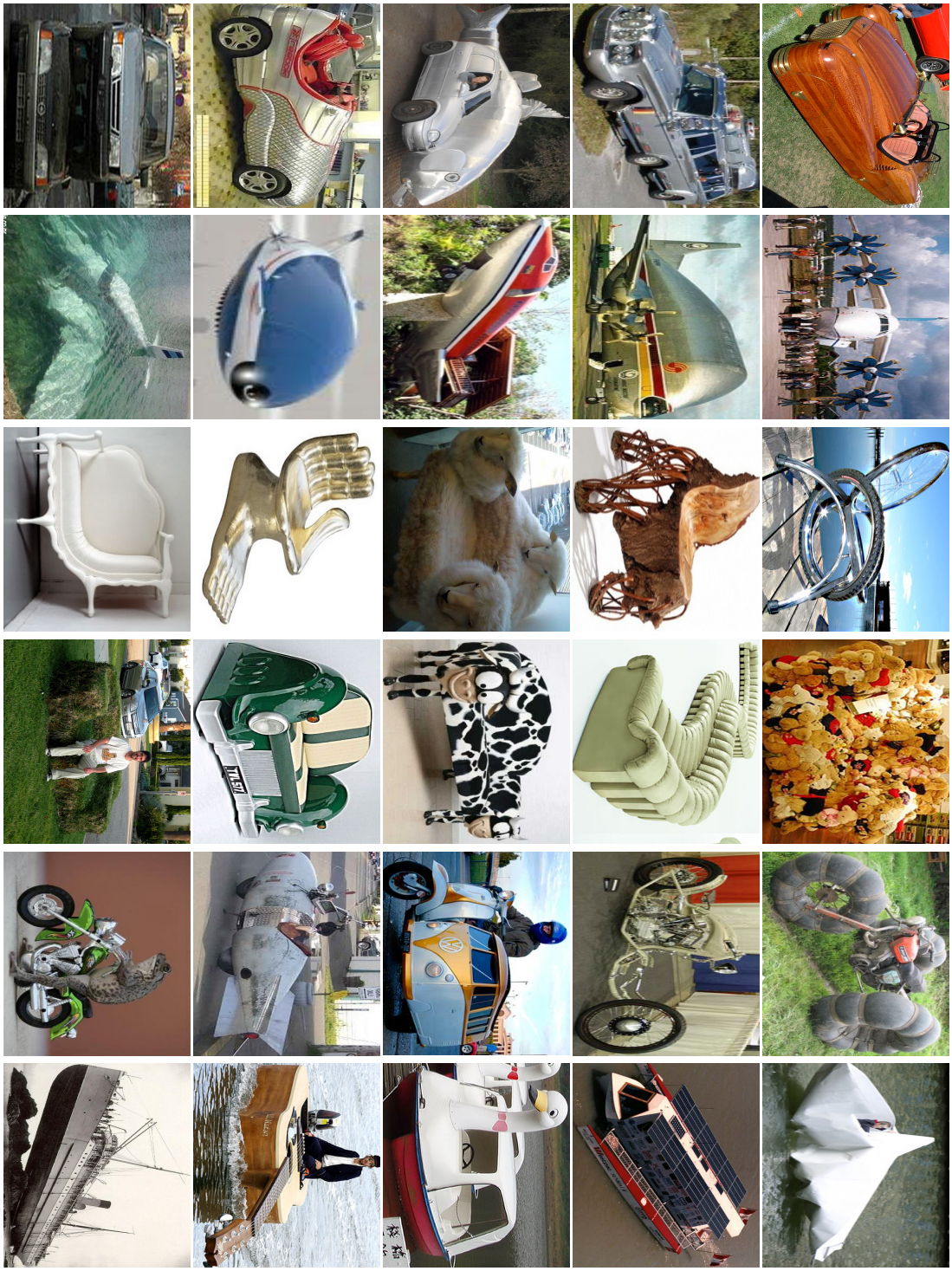

Figure 1.1: Examples images of our “1001 Abnormal Image” dataset. Images in each column correspond to one object category (cars, airplanes, chairs, sofas, motorbikes, boats). Despite the limited number of examples, these images show a variety of abnormality reasons observed in our collection

intelligent system that can detect abnormalities. We argue that abnormality detection could improve learning object categories based on their attributes. Abnormality recognition in images has been widely used in surveillance systems. There has been recent interest in investigating what should be reported as an output of a recognition system [54]. When describing an image, humans tend not to mention the obvious (simple category memberships) instead to report what is worth mentioning about an image. We argue that abnormalities are among major components that form what is worth mentioning. We have probably heard statements like "look at that furry dog," "this is a green banana," several times. This type of reasoning is possible via the proposed framework of abnormality recognition. We form category structures in terms of common attributes in the category and reason about deviations from categories in terms of related attributes. Our method acknowledges category memberships for atypical examples and reports its reasoning behind any abnormality detection.

1.2 Challenges of Abnormality Detection in Images

There are several issues and concerns in abnormality detection:

- *First*, researchers are not in an agreement about what is a typical sample of a category and what makes humans distinguish typical instances from atypical ones [133]. The definition of abnormality in the visual space is even more complex. For example, there is no general rule as what is a typical car. Even if there were such a rule, it might vary across people and categories.

- *Second*, abnormality (atypicality) in images is a complex notion that happens because of a diverse set of reasons that can be related to shape, texture, color, context, pose, location or even a combination of them. Figure 1.1 shows large variability of abnormality reasons among examples of images of six object categories that human subjects denoted as abnormal.

- *Third*, there is a gradual transition from typical to atypical instances, so simple discriminative boundary learning between typical and atypical instances does not seem appropriate. Fourth, with the limited number of abnormal images it is hard to be comprehensive

Method	Top-1 error (%)	Top-5 error (%)
AlexNet [89]	74.96 (38.1)	47.07 (15.32)
OverFeat [148]	75.62 (35.1)	46.73 (14.2)
Caffe [80]	77.12 (39.4)	46.86 (16.6)
VGG-16 [152]	77.82 (30.9)	47.49 (15.3)
VGG-19 [152]	76.35 (30.5)	45.99 (15.2)

Table 1.1: State-of-the-art Convolutional Neural Networks (trained on normal images) fail to generalize to abnormal images for the task of object classification. Numbers in parenthesis show the reported errors on normal images (ILSVRC 2012 validation data), while numbers next to them is the error on our abnormal images.

with all aspects of abnormality. This suggests that computational models for identifying abnormalities should not rely on training samples of abnormal images. This is also aligned with how humans are capable of recognizing abnormal images while only observing typical samples [134].

1.3 Contributions

The goal of this thesis is to extract a list of reasons of atypicality, enumerating distinct modes or types of abnormal images, and to derive computational models motivated by human abnormality classification. The contribution of this thesis is manifold: We conduct a human-subject experiment to determine a typology of images judged abnormal by human observers and collect data that facilitates discovery of a taxonomy of atypicality. Analysis of the data lead us to a coarse taxonomy of three reasons for abnormality: .

- Collecting and publicizing the largest dataset of annotated abnormal images. We published a dataset of abnormal images that is the largest in terms of both the number of images and the variety of reasons of abnormality present in images.

- Conducting human-subject experiments to determine a typology of images judged abnormal by human observers. These experiments investigate both images and objects that look strange, providing data that helped us to infer a systematic analysis of reasons of abnormality in images.

- Comprehensive analysis of abnormality reasons and proposing a taxonomy of abnormality reasons in images. We inferred a full taxonomy of abnormality reasons and found hidden factors that generate these reasons.

- Designing and implementing computational models for three main reasons of abnormality: object-centric, scene-centric, and contextual.

- Learning visual classifiers for unseen classes from pure textual descriptions, where it helps expanding the abnormality detection framework to cover categories that are not present in training. Building localized attribute classifiers that provide us with more information about abnormality cues in objects and images.

- Improving the stat-of-the-art object classifiers via using typicality signals of training images. We train Convolutional Neural Networks (CNN) with an extended generalization capacity that can classify extreme cases of abnormal objects.

1.4 Acknowledgments and Referred Publications

The research material in Chapter 2 was produced in collaboration with Dr. Ahmed Elgammal, Dr. Jacob Feldman and Dr. Ali Farhadi. This collaboration resulted in the following publication and presentations:

- [146]: Babak Saleh, Ali Farhadi, Ahmed Elgammal. ”Object-Centric Anomaly Detection by Attribute-Based Reasoning” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013.

- [145]: Babak Saleh, Ahmed Elgammal, Jacob Feldman. ”Toward a Taxonomy and Computational Models of Abnormalities in Images” The Thirtieth AAAI Conference on Artificial Intelligence (AAAI) 2016.

- [138]: Babak Saleh. ”Wow! that looks strange: computational models for detection and reasoning about abnormalities in images” ACM AI Matters 2016.

The research material in Chapter 3 was produced in collaboration with Dr. Ahmed Elgammal, and Dr. Ali Farhadi. This collaboration resulted in the following publication and

presentations:

- [146]: Babak Saleh, Ali Farhadi, Ahmed Elgammal. ”Object-Centric Anomaly Detection by Attribute-Based Reasoning” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013.
- [138]: Babak Saleh. ”Wow! that looks strange: computational models for detection and reasoning about abnormalities in images” ACM AI Matters 2016.

The research material in Chapter 4 was produced in collaboration with Dr. Ahmed Elgammal, Dr. Jacob Feldman and Dr. Ali Farhadi. This collaboration resulted in the following publication and presentations:

- [146]: Babak Saleh, Ali Farhadi, Ahmed Elgammal. ”Object-Centric Anomaly Detection by Attribute-Based Reasoning” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013.
- [145]: Babak Saleh, Ahmed Elgammal, Jacob Feldman. ”Toward a Taxonomy and Computational Models of Abnormalities in Images” The Thirtieth AAAI Conference on Artificial Intelligence (AAAI) 2016.

The research material in Chapter 5 was produced in collaboration with Dr. Ahmed Elgammal, and Dr. Jacob Feldman. This collaboration resulted in the following publication and presentations:

- [143]: Babak Saleh, Ahmed Elgammal, Jacob Feldman. ”Incorporating Prototype Theory in Convolutional Neural Networks” Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI) 2016.
- [144]: Babak Saleh, Ahmed Elgammal, Jacob Feldman. ”The Role of Typicality in Object Classification: Improving The Generalization Capacity of Convolutional Neural Networks” arXiv 2016.

The research material in Chapter 6 was produced in collaboration with Dr. Ahmed Elgammal, and Mohamed Elhoseiny. This resulted in the following publication and presentations:

- [48]: Mohamed Elhoseiny, Babak Saleh, Ahmed Elgammal. "Write a classifier: Zero-shot learning using purely textual descriptions." Proceedings of the IEEE International Conference on Computer Vision, 2013.

- [47]: Mohamed Elhoseiny, Ahmed Elgammal, Babak Saleh, "Write a Classifier: Predicting Visual Classifiers from Unstructured Text Descriptions", arXiv, 2016

During the course of my PhD studies, I had the privilege of collaborating with Kanako Abe, Ravneet Arora, Mira Dontcheva, Aaron Hertzman, and Zhicheng Liu. Following publications are the results of these collaborations:

1. [142]: Babak Saleh and Ahmed Elgammal. "Large-scale Classification of Fine-Art Paintings: Learning The Right Metric on The Right Feature" Journal of Digital Art History 2016.

2. [139]: Babak Saleh, Kanako Abe, Ravneet Arora, Ahmed Elgammal. "Toward automated discovery of artistic influence" Multimedia Tools and Applications 2016.

3. [142]: Babak Saleh and Ahmed Elgammal. "A unified framework for painting classification" IEEE International Conference on Data Mining Workshop (ICDMW) 2015.

4. [141]: Babak Saleh, Mira Dontcheva, Aaron Hertzmann, Zhicheng Liu. "Learning Style Similarity for Searching Infographics", 41st annual conference on Graphics Interface (GI) 2015.

5. [46]: Ahmed Elgammal and Babak Saleh. "Quantifying Creativity in Art Networks" International Conference on Computational Creativity (ICCC) 2015.

6. [140]: Babak Saleh, Kanako Abe, Ahmed Elgammal. "Knowledge Discovery of Artistic Influences: A Metric Learning Approach" International Conference on Computational Creativity (ICCC) 2014.

Chapter 2

Learning a Taxonomy for Abnormality in Images

The taxonomy of abnormality in images is not well-defined either in Psychology or Computer Vision. We design a human subject experiment to discover a coarse taxonomy for abnormality. To this end, we first need to collect a dataset of abnormal images that is more comprehensive than what has been used in prior work. Then we implemented several human subject experiments to investigate how humans perceive abnormality in images. We ran unsupervised analysis of subjects' responses to learn hidden factors of their decision making and propose a taxonomy of reasons of abnormality.

2.1 1001 Abnormal Image Dataset

For the purpose of our study, we needed to collect an exploratory dataset of abnormal images. We believe no such dataset exists in the computer vision community. There are datasets for studying abnormal activities in videos, however our goal is to study abnormalities in images. To be in line with the image categorization research we chose object classes from PASCAL dataset [50] to build our dataset. To collect the abnormal images in our dataset, we used image search engines, in particular Google images and Yahoo images where we searched for keywords like "Abnormal", "Strange", "Weird" and "Unusual" in combination with class labels like cars, airplanes, etc. The top results from the search engines were pruned by removing duplicates, obviously irrelevant images and very low quality pictures. Unlike typical images, it is not that easy to find abundance of abnormal images. As a result we narrowed down the object classes to only six classes of PASCAL where we could collect at least 100 images: namely "Airplane", "Boat", "Car", "Chair", "Motorbike" and "Sofa". The overall data set contains 617 images. The collected images were annotated by marking a bounding box around the salient object in each image.

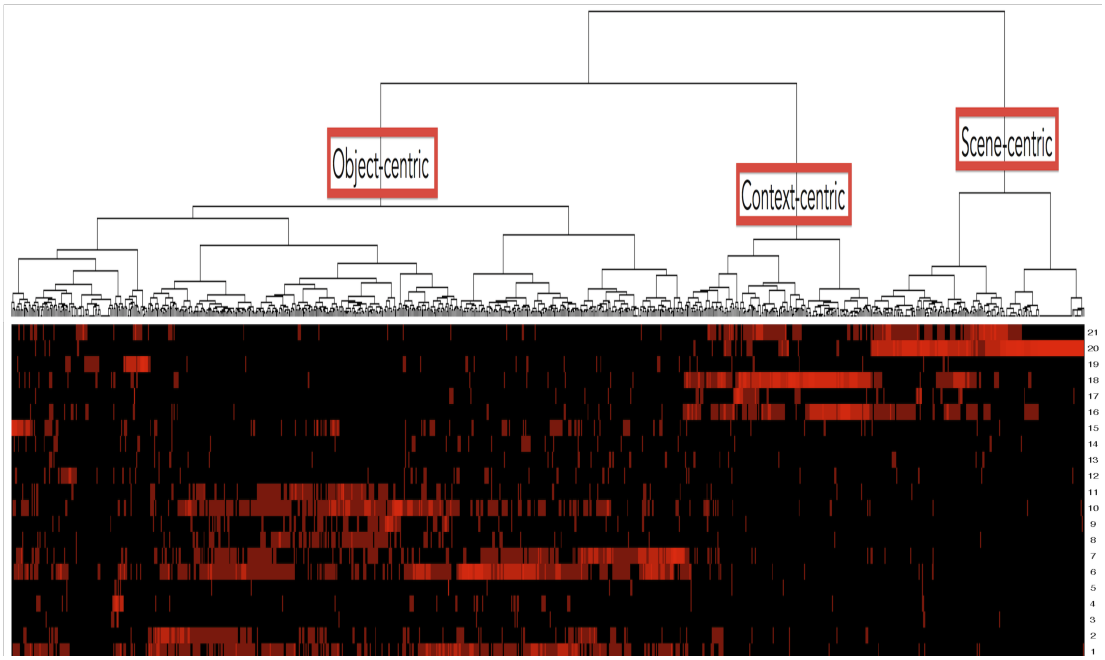

Figure 2.1: Agglomerative clustering of abnormal images (columns) based on the human subject responses for each abnormality reason (rows). The dendrogram on top of the figure shows how the abnormal images in our dataset can be grouped to make three clusters, reflecting three major latent categories of abnormality. Each cluster corresponds to a specific list of abnormality reasons. Details of these three categories of abnormality can be found in Table 2.1

Previous datasets for abnormality research are either not specifically designed for the task of abnormality detection [110], or limited to a specific type of abnormality [[110, 122] are focused on contextual cues, and [146] is concerned with object-centric reasons)]; or of small size [[110] has 40 and [122] has 200 images].

In order to study abnormalities in images with more details, this paper introduces a dataset that is more comprehensive both in terms of the number of images and types of abnormality. To collect this dataset, we started by gathering images from three public datasets used in [110, 122, 146], which we call "initial collection" and almost doubled the size by adding more images from the web. Our image collection process is similar to [146], but textual queries that we used for image search are not limited to abnormal objects. For examples, we used "strange street" or "weird living room" as additional queries. After downloading a large number of images, we pruned the result by removing duplicates and very low-quality images. Then we merged these images and "initial collection" into the final dataset with a total number of 1001 unique

Main Category	Detailed Reasons in Amazon Mechanical Turk Experiment
Scene-centric	Strange event happening in the scene(21); Strange scene(20)
Context-centric	Atypical object size(19); Strange location of the object(18); Atypical object pose(17); Weird combination of objects and scene(16)
Object-centric	Unexpected part(7); Weird shaped part(6); Misplaced part(5); Missing part(4); Body posture(14); Mixture of object classes(13); Un-nameable shape(12); Object is not complete(3); Unknown object(15); Object in the shape of another object(2); Atypical pattern(10), Weird color(9), Strange material(11), Weird texture(8); Strange object contour(1)

Table 2.1: Learned taxonomy for reasons of abnormality in images based on our human subject experiment. Numbers in the parenthesis are indexes of reasons, which correspond to the rows in Figure 2.

abnormal images. Figure 1.1 shows some images of this dataset. We validated our collection and acquired image annotations by conducting a human-subject experiment as explained next.

2.2 Human Subject Experiment

2.2.1 First Round - Object Level

The subject of abnormality is rooted in people's opinion, so any work on detecting strange images without any comparison to the human decision is not informative. There are other multiple reasons that motivates studying human subjects' responses to our collected images. 1) Validating our collected dataset. 2) Providing ground truth 3) Providing some insight about how people judge about the abnormality of images.

Therefore, we designed a preliminary survey for human subjects and we used Amazon Mechanical Turk to collect people responses. Given an image with a bounding box around the most salient object, subjects were asked following questions. First, the subjects were asked whether the image seems to be normal or abnormal. If the subject decided that the image is abnormal, the following questions were asked where multiple selections are allowed: 1) Which category best describes the object, from a list of the six categories in our dataset. 2) Whether abnormality is because of the object itself or its relation to the scene. 3) Rate the importance of each of the attributes in affecting their decision about normality (Color, Texture/Material, Shape/Part configuration, Object pose/viewing direction) 4) Also the subjects were asked to

Figure 2.2: Statistics of abnormality reasons (Shape, Texture, Posture, and Color) in human responses for abnormal objects in our dataset.

comment about context abnormality if it is the case.

Figure. 2.4 shows the subjects' average rating for the different causes of abnormality for each category. This is for the images that subjects decide that the abnormality stems from the object itself. The figure clearly shows that in all categories atypical shape is the most common cause of abnormality, followed by texture/material, then pose and color. Interestingly this trend is independent of the category of the object.

Figure. 2.2 illustrates the variability of responses for each category of objects, based on four reasons of abnormality. We observe that except for the airplane category, the variances in ratings for each cause of abnormality is relatively small. The rating for the airplane has a large variance, which might indicate that the real reason for abnormality is not one of the four given reasons. We conclude that the variation of responses, or equivalently, agreement between all annotators for abnormality judgment, is dependent on the object category. We hypothesize that these dependencies are related to the familiarity of human subjects with the object category of interest. For example, compare to chairs or sofa airplanes are less frequently seen in everyday life.

	Aeroplane	Boat	Car	Chair	Motorbike	Sofa	None
Aeroplane	908	10	7	1	0	0	51
Boat	62	868	57	0	1	1	44
Car	7	9	1072	3	0	1	52
Chair	0	1	11	861	1	166	36
Motorbike	17	0	31	3	540	0	38
Sofa	1	1	3	273	0	666	86

Figure 2.3: Confusion matrix for the task of object classification in Human subject experiment.

Figure 2.3 represents the confusion matrix for the human subjects in deciding the category of the object. An important conclusion from this study is that the variance in subjects' decisions about normality/abnormality is much less than the variance in their decisions about the object categories. Simply it shows that users might not agree on category of the object, but with high confidence they concur the judgment about the abnormality of the object.

2.2.2 Second Round - Image Level

We conducted a two-phase experiment. First, we asked four subjects to take an on-site test and exposed each subject to a unique set of images from our dataset. The human subject was asked to determine whether the images are abnormal, and if they are, to explain the reason behind the abnormality in their own words. The goal of this step is to compile an initial comprehensive list of reasons for abnormality in images.

We enumerated the responses into a list, and did not merge them unless two reasons clearly refer to the same notion (e.g. "This object does not have an expected part" and "One part is missing for this object" are classified as the same reason). By this process we came up with a list of 21 fine-grained reasons for abnormality written in plain English. Some example reasons include "An unexpected event is happening in this image", " Weird object material" and "Missing part for the object". The full list of fine-grained reasons is shown in Table 2.1. We denote this list by "expanded abnormality list". We understand that this list might not be universal for all possible reasons of visual abnormality, but we believe it covers most types of abnormalities in our dataset.

In the second phase, our goal was to annotate all images in our dataset with a reasonable number of human subject responses, and discover a hierarchy of these reasons via an unsupervised approach. In order to complete this large-scale experiment, we asked annotators on

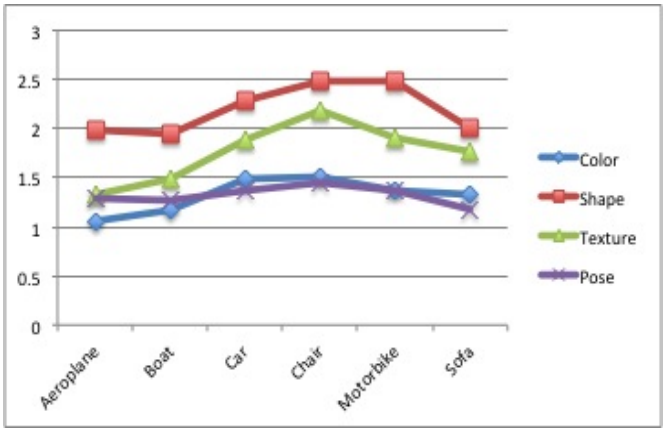

Figure 2.4: Rating of four main reasons of abnormality for six classes of objects. Independent of object category, shape is the most important reason that can make an object look abnormal.

Amazon Mechanical Turk to annotate images in our dataset based on the 21 reasons in the expanded abnormality list. Also we added two extra choices: *"This image looks abnormal to me, but I cannot name the reason"* or *"Abnormal for a reason that is not listed"* followed by a text box that the subject could write in it. This gives annotators the opportunity of describing the abnormality in their own words.

As one image can look abnormal because of multiple cues, annotators could select multiple reasons and were not limited to the 21 reasons on the list. We picked annotators with a good history for the task of image annotation and categorization (users for whom at least 95% of previous responses over the past three months were accepted). Subjects could not take a task twice and for each image we aggregated responses from six unique human subjects. To verify the quality of annotations, we randomly asked annotators to take an image for the second time to see if their response matched his/her previous response. Due to the importance of non-random responses, if an annotator showed a random behavior in choosing the reasons of abnormality, we re-sent the task to the rest of participants and stopped the suspicious annotator from taking future HITs. In total 60 unique human subjects participated in this experiment.

First Factor	0.891447558743145 0.748373028770638 0.694005373675896 0.644901858617892 0.0790435014872081	“strange texture” “strange material” “atypical pattern” “atypical color” “Object in shape of another object”
Second Factor	0.958508457108101 0.535534222059008 0.331676465239351 0.0531447346520316 0.0322912279988133	“missing part” “Object is not complete” “misplaced part” “body posture” “un-nameable shape”

Figure 2.5: The first two hidden factors of abnormality that generate 15 fine-grained reasons in object-centric cluster of abnormal images. Red color indicates more important reasons (right column) for each inferred factor (left column).

2.3 Discovering a Taxonomy of Abnormality:

We averaged the responses across all subjects for every image and for each of the 21 reasons. This results in a an embedding of the images into a 21-dimensional space, i.e. each image is represented with a 21-dimensional response vector. We hypothesize that there is a latent abnormality subspace (space of reasons for abnormality in images); and measuring similarity between the response vectors for images is expected to reflect the similarity between them in the latent abnormality space. To discover a taxonomy of abnormality, we performed unsupervised learning on the collection of response vectors using bottom-up agglomerative clustering. We used the Euclidean distance as a dissimilarity measure and the Ward’s minimum variance criteria [109] for the linkage function. At each step, a pair of clusters that result in the minimum increase in the within-cluster variance are merged.

Figure 2 shows the resulted dendrogram of images and the corresponding responses in the 21-dimensional space. One can spot three main clusters in this dendrogram, which directly corresponds to grouping of reasons of abnormality. The implied grouping is shown in Table 2.1. Consequently, we can name intuitive atypicality groups based on this coarse taxonomy: Scene-centric atypicality, Context-centric atypicality, and Object-centric atypicality. We performed several experiments on clustering with different linkage functions and metrics; however, we

First cluster:	Second cluster:
"atypical pattern"	"weird-shaped parts"
"strange material"	"Strange contour (boundary)"
"strange texture"	"unexpected part"
"weird-shaped parts"	"Object in shape of another object"
"atypical color"	"missing part"
"Strange contour"	"body posture"

Figure 2.6: Using factors inferred in the first step, original responses are transformed and images are clustered. Each column lists all reasons that are prominent in each cluster. Color codes shows the importance of each reason for that particular cluster.

observed that this coarse taxonomy is robust over changes in the clustering parameters. It is interesting that prior research is broadly consistent with this taxonomy: the work of [122, 110, 45] proposed models to predict contextual atypicality, and proposed models of [146] predict object-centric abnormality. Thus we conclude that our taxonomy, which is motivated by human judgments, encompasses previous approaches in a more systematic way.

2.4 Scrutinizing the Taxonomy of Abnormality

The taxonomy of abnormality reasons that we inferred from hierarchical clustering of human subject experiments will be the basis of our computational models for detecting abnormality in images. However, before moving forward with this taxonomy, we conducted a thorough analysis to analyze its robustness. Additionally, as the main component of this taxonomy is devoted to the object-centric abnormality, we scrutinize the cluster of images that are labeled as abnormal because of having an abnormal object. In this section, we propose a principled approach to discover underlying common grounds of humans' responses (e.g. three main reasons of abnormality listed in the previous section). This approach provides us with further analysis of each one of these factors in more details.

We use "Factor Analysis" models to learn hidden factors that generate 15 fine-grained object-centric reasons of abnormality. The hypothesis behind taking this approach is as following. Although the diversity of fine-grained reasons of abnormality gives us the opportunity

of having semantically meaningful reasons generated by human subjects, but it is very likely that some of these reasons are referring to the same visual cues, but maybe with different languages. This issue is mainly rooted in subjects' biases, rather than language barriers. As a result, it cannot be resolved via disambiguation techniques used in natural language processing community. Instead, we use explanatory factor analysis to learn some hidden factors that generate these fine-grained reasons of abnormality. We hypothesize that more abstract reasons increases the agreement between subject responses. However, we intentionally let users to generate fine-grained reasons to benefit from more informative ground truth data.

The detail of the proposed framework for analyzing responses is as following. By running explanatory factor analysis, we find number of factors that are statistically significant for expressing the current list of fine-grained reasons. Based on these inferred factors we project the original responses and cluster all images when the projected responses are used as input features. For each cluster we iterate the aforementioned steps.

We applied this procedure for the group of images that fall into the cluster of object-centric abnormality, and found that we only need two hidden factors to generate 15 fine-grained reasons that correspond to object-centric abnormality. Figure 2.5 lists these reasons (last column) along with their explanation factors (middle column). Interestingly, this mutual excursive grouping of reasons is semantically meaningful as well.

We projected the raw subjects' responses (to 15 reasons of object-centric abnormality) into a two dimensional space of hidden factors by using the projection matrix. Next, clustered (via K-means) images in this projected space. Figures 2.7& 2.8 show samples images of the first cluster and Figures 2.9& 2.10 show some samples of the second cluster. By looking at samples of each cluster, we can make the following claim. While the second cluster contains images that present abnormal objects mainly because of their shape characteristics, the first cluster has objects that look abnormal due to some other reasons rather than shape (e.g. strange texture).

In order to evaluate this hypothesis, we averaged the responses for original 15 fine-grained reasons of abnormality over all images of each cluster. Figure 2.6 enumerates the reasons for each cluster, when they are ranked based on the average responses for images. Interestingly, these lists of sorted abnormality reasons confirm our hypothesis about the underlying notion of abnormality within each cluster based on hidden factors.

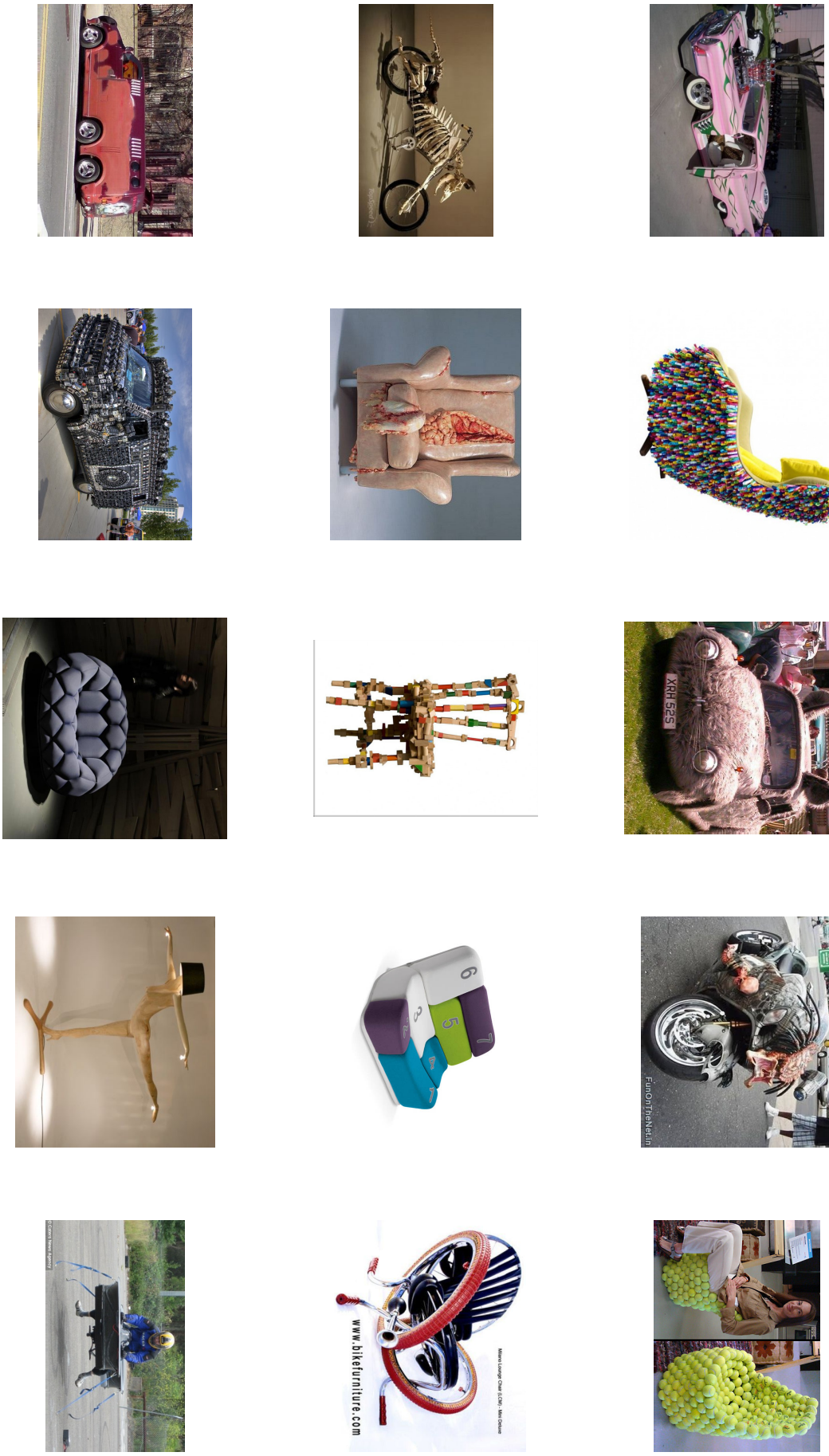

Figure 2.7: Sample images of the first cluster of object-centric abnormality after factor analysis. Abnormality of objects within this cluster mostly resonates with the texture and material related

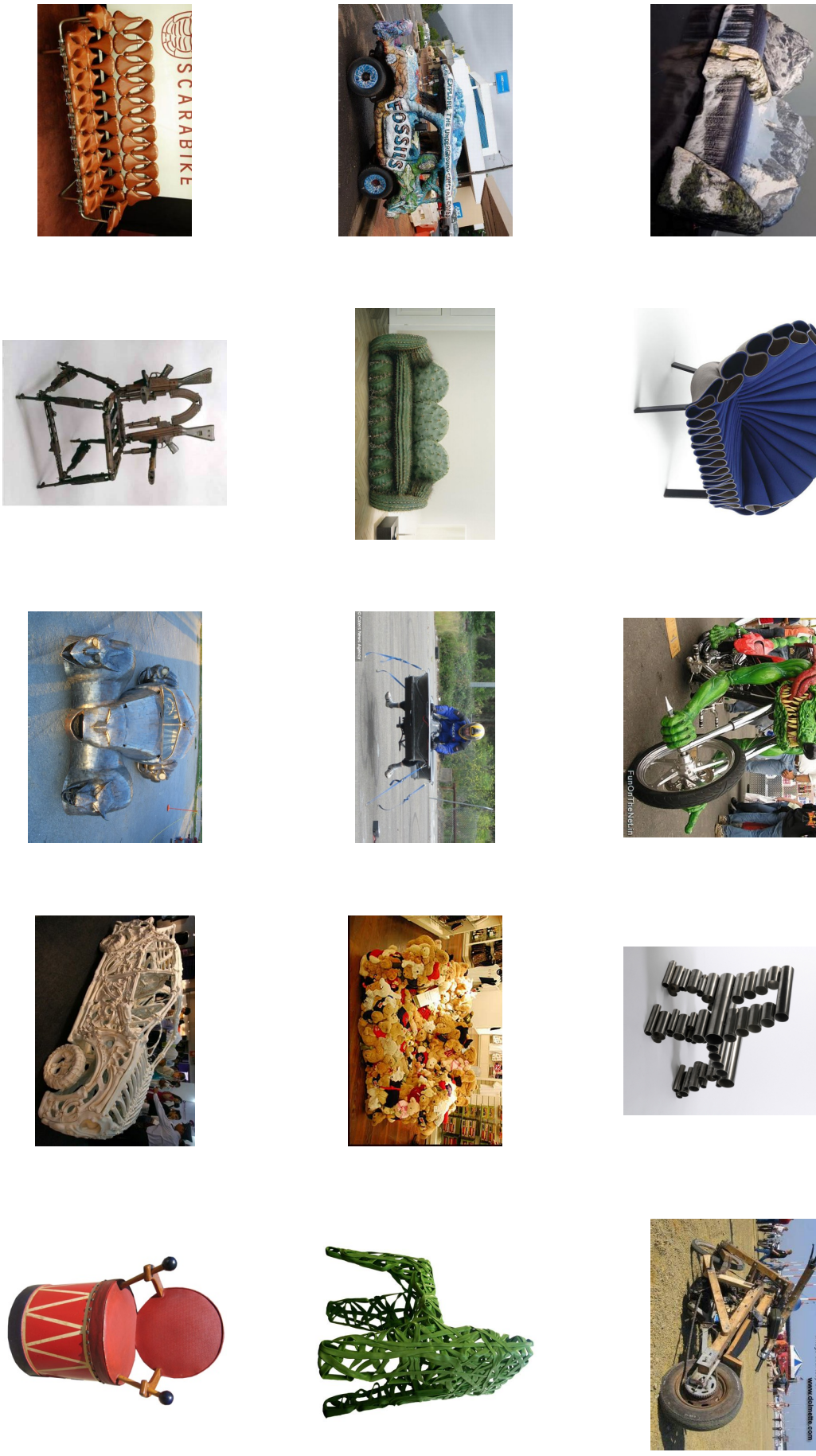

Figure 2.8: Sample images of the first cluster of object-centric abnormality after factor analysis. Abnormality of objects within this cluster mostly resonates with the texture and material related

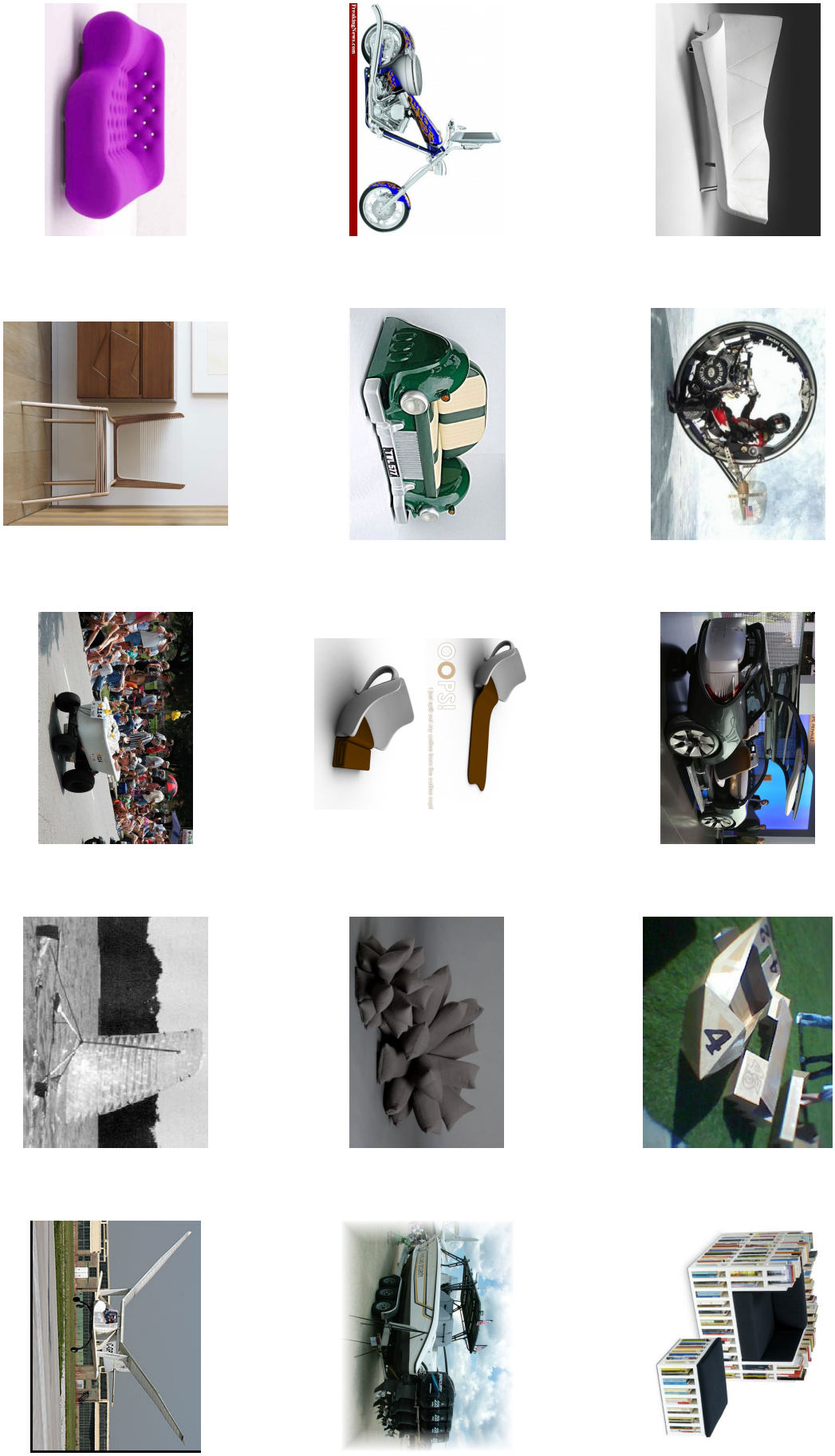

Figure 2.9: Sample images of the second cluster of object-centric abnormality after factor analysis. Images of this cluster represent objects that look abnormal due to weird shapes or

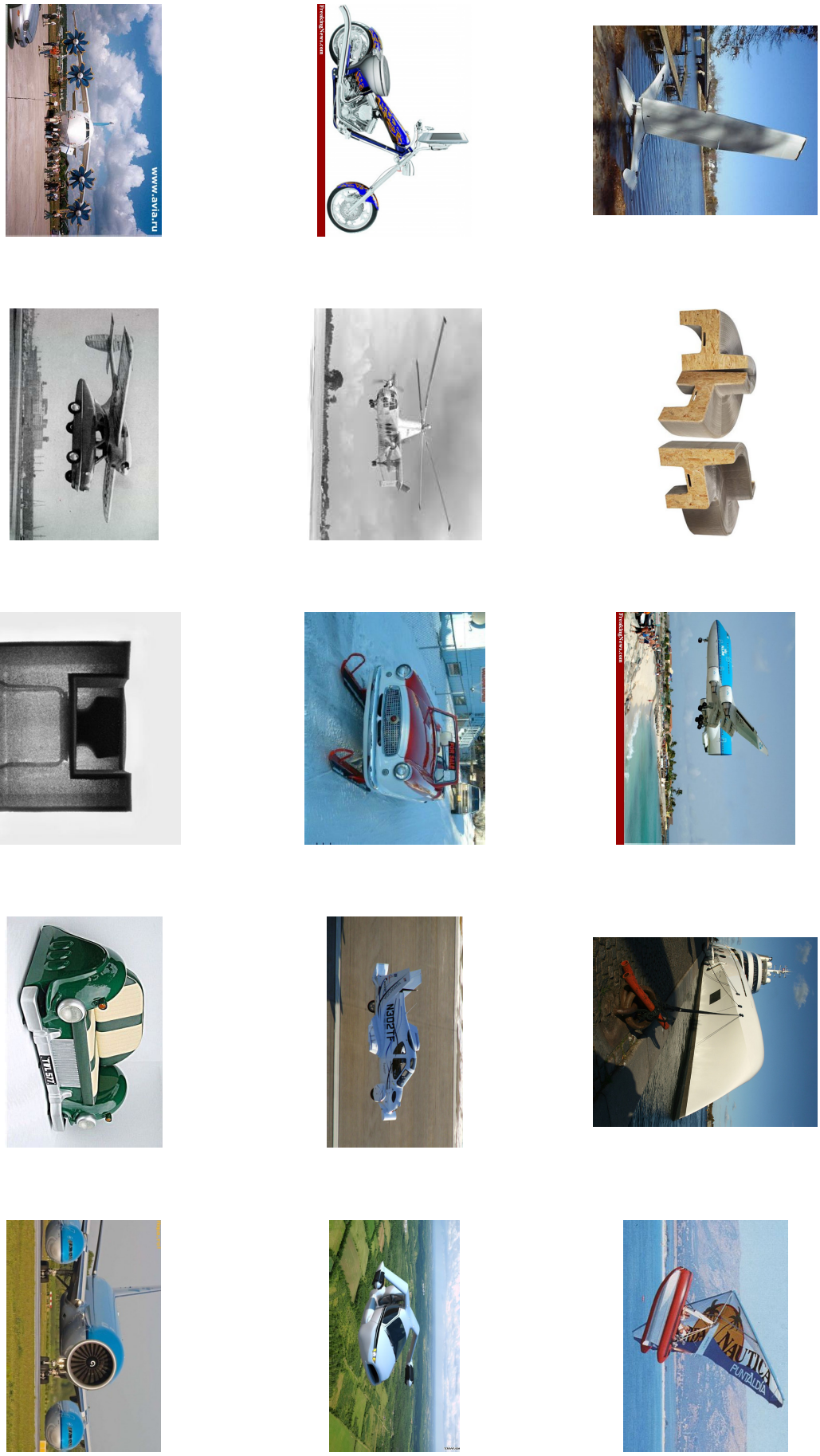

Figure 2.10: Sample images of the second cluster of object-centric abnormality after factor analysis. Images of this cluster represent objects that look abnormal due to weird shapes or

Chapter 3

Modeling Normality and Its Connection to Measuring Surprise

We model abnormality as meaningful deviation from normality. This leads us building computation models of normality in a way that : 1) Can be trained without observing abnormal images, 2) Deviation from these models -center of the population- is meaningful and can be used for detecting abnormal objects.

In this section , we first talk about classification of objects and its relation to finding abnormal objects. We argue that object classifiers should be able to not only categorize normal objects (ones that are far from classification boundary and close to the center of category), but also put abnormal objects in relatively appropriate locations. We believe generative models of object classification are good fit for this purpose. Next, we develop different approaches to build such models of normality, and investigate probabilistic frameworks -as our final approach- in more detail. We develop our models using the notion of visual attributes, which empowers our model with reasoning about abnormal objects.

3.1 Classification Paradigm and Abnormal Objects

There are two observations that motivate our abnormal object classification model, based on soft-assignment of category memberships. First, the common approach to multi-class recognition involves performing several one-versus-all classification/detection tasks. Such a discriminative paradigm shows superiority in categorization and thus widely used. This implies that there is an assumption about the existence of a clear boundary between object categories. Taking abnormalities into the consideration, these boundaries between basic level categories become not as clear. In particular, objects in group III & IV in the abnormality taxonomy, contain several features and attributes that are common in multiple classes. It might be hard or impossible to identify the correct category of these objects solely based on visual features

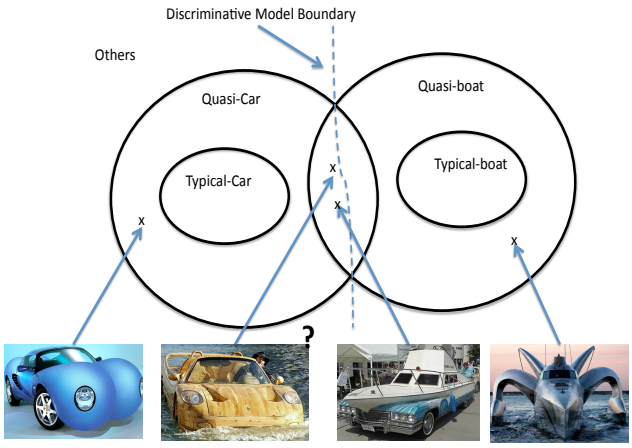

Figure 3.1: Confusion on categorization of abnormal objects caused by fuzzy membership of these objects.

or attributes of these objects themselves. Further investigations of the scene context and the functionality of these objects are necessary to determine the true category of these objects. Therefore the outcome of the categorization phase in a recognition system should not be a hard category assignment, rather membership scores of different categories.

Second There is a fundamental difference between our definition of abnormalities and the existing definitions in the literature. In conventional definitions, unusual examples are the ones that are not similar to any (or similar to very few) previously known examples. However, our definition of abnormality (as discussed in chapter 1) entails a form of similarity while being different. For example, based on conventional definitions, a chair can be thought of as an atypical example of car category, atypical example of motorbike category, etc. In contrast, our definition of abnormality requires the example to be "some how" similar to some categories while being different in some related attributes.

Where do abnormal instances of categories lie in a visual feature space? The two aforementioned observations lead to the following hypothesis. For each category we define two different sets: the set of normal/typical instances, and the set of quasi-category. The set of quasi-category contains the instances that resembles the category in certain features or attributes

however they are atypical from category prototypical examples. This is illustrated in Fig 3.1 where we use two categories for simplicity, car and boat. There are the sets of typical-cars and typical-boats which are disjoint; and there are also the sets of quasi-cars and quasi-boats. The typical-category set is a subset of the quasi-category set. The quasi-category sets can intersect and do intersect in many cases. For example, there are instances that resembles cars and boats that belong to the intersection of the quasi-car and quasi-boat sets.

The typical discriminative categorization algorithms do not consider this setup, and assume a clear boundary between categories. Consequently, they are bound to be confused about instances in the quasi-category intersections. Humans also get confused about these instances as apparent form the confusion matrix in Fig 3.1. This might suggest detecting abnormal instances based on how close they fall to the margin. However, this is not sufficient since abnormal instances can also be away from the margin, anywhere in the quasi-category set.

Therefore, to be able to detect abnormal instance of category c we need some indicator that this instance is in quasi-c and not in typical-c. However, the challenge is that the boundary of typical-c and the quasi-c is not well defined, as well as the boundary between quasi-c and the rest of the world. Furthermore we should not train on abnormal instances (humans do not train on abnormal images), therefore a discriminative approach for detecting abnormality within a class is neither feasible, nor desirable.

The above discussion makes it clear that a generative model is needed to model typical instances of a given category. Our model produces a distribution over categories and avoids making hard decisions till the very end in the process. Conditioned on a category, we can decide if the observed object is a typical(normal) sample of that category or not. Basically, we determine how close is the object from the majority of normal samples. Formally, we are interested to model $P(\neg N|A) = 1 - P(N|A)$, where random variable N stands for being a normal object and A is a random variable for visual attributes. This formulation clarifies that we only need to compute $P(N|A)$ to be able to judge the abnormality of the object, without seeing any abnormal object during training. This term can be explained as an aggregation of normality scores over possible object categories: $P(N|A, C_k)$, where $C = \{C_1 \cdots, C_K\}$ is a random variable indicating the category.

Theoretically by Bayes' rule the posterior $p(N|A, C_k)$ could be achieved, and the atypicality given the category is simply the complement event. Thus we can obtain $p(N, C_k|A) = p(N|C_k, A)p(C_k|A)$. However, in our case we cannot get the posterior because we do not have a model for $P(A|\neg N, C_k)$, which is the generative model for the atypical instances. This is because we should not train on atypical instances. Therefore we have to use the likelihood to decide about typicality.

Our proposed approach models the typicalities by leveraging the hidden structures among typical examples of categories using an attribute-based representation. Once the typicalities have been modeled, abnormalities can be defined as meaningful deviations from typicalities within the category. To model this deviation one needs to encode related attributes and select accordingly. Once deviations have been formulated, our method can classify atypical examples, and reason about the rational behind any detection in terms of attributes. To achieve this goal we need to 1) investigate generative methods for discovering the structure of typicality, 2) devise methods to measure deviations from typicality.

Unlike most attribute based frameworks, our attributes are not designed to provide cross category generalization. In fact, we intentionally learn our attributes to encode inside category relationship. Because, there are subtle differences for attributes inside categories. A typical bicycle wheel is considered atypical for cars. Furthermore, patterns of occurrence of attributes may be very different for very similar categories. Later we show how to benefit from these patterns of co-occurrences.

3.1.1 Relevant Attribute Selection

An attribute is useful for detecting normality/abnormality if it is common with a given category. For example, cars typically have wheels, if a car in an image does not have wheels and there is no obvious reason for not seeing the wheels, then it is probably abnormal. On the other hand an attribute is useful for detecting abnormality if it is rarely seen in a given category. Take the car example again, a car is not expected to have wings or eyes. Existence of such attributes are a strong cue for abnormality. So the absence of common attributes or existence of peculiar attributes for each category are useful cues for detecting abnormality.

Let $A_i(x) : \mathcal{X} \rightarrow \mathbf{R}$ be the confidence of the i-th attribute obtained from the i^{th}-attribute

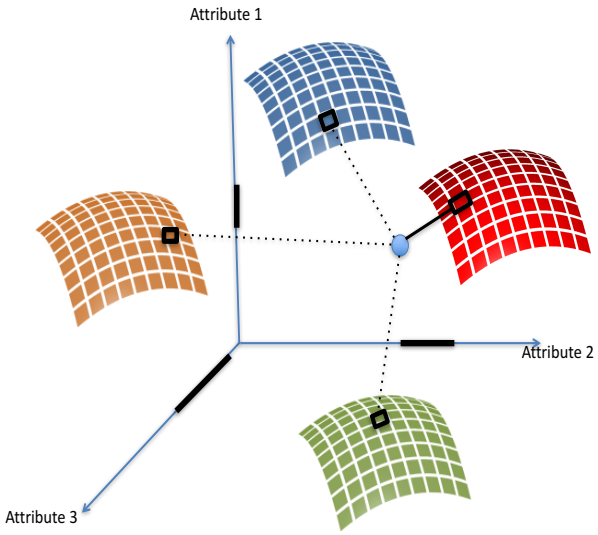

Figure 3.2: Illustration of object categories modeled by manifolds based on visual attributes.

classifier for image x. We need to model the conditional density $p(A_i|N, C_j)$ for each attribute i given typical example of category j. Both common attributes and peculiar attributes share the properties that they should have a peaky conditional densities, regardless of the value of the confidence. Therefore, we use an entropy measure to detect such attributes. We compute the conditional entropy $H(A_i|N, C_j)$ for each attribute and category pair. The lower the entropy the more peaky the distribution of the confidence over typical images, and hence the more relevant that attribute for detecting typicality/atypicality. Therefore we use $1/H(A_i|N, C_j)$ as the typicality/atypicality-relevance measure of attribute i for category j.

3.2 Modeling Typicality

For modeling typicality we need to learn generative models in terms of the conditional class densities $p(x|N, C_k)$. We use an attribute space for that purpose, i.e. we need to model $p(A_1(x), \cdots, A_M(x)|N, C_k)$, where M is the number of attributes. We investigated several models of typicality, which we will summarize in this section.

Modeling typicality manifold: In this approach we hypothesize that typical images lie on a low-dimensional manifold in the attribute space. As Figure 3.2 illustrates this model based

on three visual attributes (three axis): We explicitly model that typicality manifold for each category (colored surfaces) and compute deviation from abnormality by measuring the distance of a test image (blue dot) each category. Given a test image we find its nearest neighbor from the training data of a given category and then compute the perpendicular distance to the tangent space of the manifold at that point. This can be achieved by projecting the test image to a local subspace for the manifold patch around the nearest neighbor point. There are two probability models for the distance to the manifold that we investigated: 1) a global Gaussian model for the whole manifold, 2) a local Gaussian model at each patch of the manifold. There are two parameters for this model, the patch size, k and the local subspace dimensionality d.

Naive Bayes' Model: In this approach we model the density $p(A_1(x), \cdots, A_M(x)|N, C_k) = \prod_i p(A_i(x)|N, C_k)$ where we use a Gaussian model for each attribute density : $p(A_i(x)|N, C_k) \sim \mathcal{N}(\mu_i^k, {\sigma_i^k}^2)$.

Manifold-based density model: This approach is similar to the Naive Bayes' Model, however instead of computing the densities $p(A_i(x)|N, C_k)$ globally, these densities are computed locally for patch of the typicality manifold. The rational is each part of the typicality manifold is expected to have different distribution.

Nonparametric Model: In this approach we model each conditional class density using kernel density estimation, i.e., we achieve an estimate of the density in the form $\hat{p}(A_1(x), \cdots, A_M(x)|N, C_k) = \frac{1}{M'} \sum_{j=1}^{M'} \prod_{i=1}^{M} g(A_i(x) - A_i(x_j))$, where $g(.)$ is a kernel function and $\{x_j\}$ are training images of class k. Here we use the kernel product, which is typically used to approximate multivariate densities.

One-class SVM: One-class SVM is widely used for estimating regions of high density. Given typical examples for each class in the attribute space, one-class svm is used to estimate a boundary of volume of high density, which can be used to detect deviations from the center of the category.

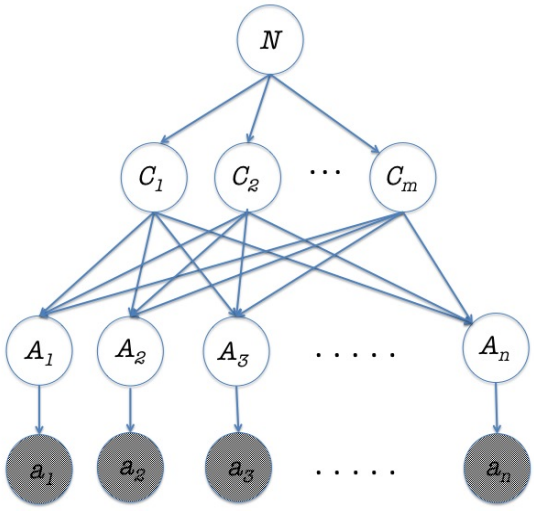

Figure 3.3: Graphical Model of Normal Objects Based on Object Categories and Visual Attributes

3.2.1 Graphical Model

Between the aforementioned models, the theoretical basis of probabilistic frameworks favor our situation more strongly. Also based on our experimental results they outperform other approaches. As a result, we develop our final framework based on probabilistic graphical models. In this model, the class membership for normal objects can be viewed as a unimodal distribution in the space of class-likelihood. This distribution for a normal image will be peaky around the correct object class and takes low value for all other classes. Normal objects of each class impose characteristic distributions over visual attributes. This means normality affects the class distribution and consequently attribute distributions through classes. This suggests modeling these dependencies with a graphical model depicted in Figure 3.3. The normality generates a distribution over classes, they consequently generate distributions over attributes. Finally, attributes generate distributions over features. In this model, $A_1, \cdots, A_N$ denote that attribute random variables, which in turn give rise to the observed image features.

At inference, our task is to figure out if a given image contains an abnormal object or not. This means that we can infer the $P(N|A)$ and use its complement to reason about abnormality: $P(\neg N|A) = 1 - P(N|A)$, where A denote the joint attribute distribution. We infer $P(N|A)$ as follows: using Bayes' rule we can write $P(N|A) = P(A|N) * P(N)/P(A)$. The joint

attribute likelihood $P(A|N)$ can be estimated by marginalizing over categories; $P(A|N) = \sum_j P(A|C_j, N)P(C_j|N)$. Conditioned on categories attributes become independent, meaning that $P(A|N) = \sum_j \prod_{i=1}^{k} P(A_i|C_j)P(C_j|N)$.

In the model we treat attributes as observable variables which are the outcomes of a calibrated discriminative attribute classifiers [53], which transfers the attribute classifier confidences to a real normalized score between 0 and 1. The attribute value given each category typically looks like a normal distribution. Therefore, we use a Gaussian distribution to model the response of each attribute classifier for each given object category. This gives us a model for $P(A_i|C_j) \sim \mathcal{N}(\mu_{ij}, {\sigma_{ij}}^2)$, where we can learn the parameters using Maximum Likelihood Estimation given training data. By inferring $P(N|A)$ one can make predictions about normality/abnormalities of given images.

3.2.2 Information Theoretic Treatment

Notion of abnormality is directly related to the concept of rareness and surprise. Imagine you are driving a car and your child in the back seat is telling you what he sees from the rear window. You will not be surprised if he tells you things like, "I see a car with wheels" or "I see a car with wiper blades". However, if he tells you "I see a car with wings" or "I see a car with fur" you will be surprised and intrigued to check it out. This relation between surprise and rareness directly motivates the use of information-theoretic formulation, inspired by the graphical model of Figure 3.3.

Given the response of an attribute classifier, we can measure the information content that response, a, given a category by $[I(A_i = a|C_j, N) = -\log P(A_i = a|C_j, N)]$ The information content will be a direct indication of rareness of observing the response value a given the learned distribution $P(A_i|C_j, N)$.

In this formulation, certain facts about attributes are ignored. Attributes are not equally relevant to each category. Different attributes play different roles within one class of objects to another. While having "Wheel" is an important factor in modeling "Cars", it is not a prominent attribute for a "Sofa". We use the inverse of the conditional entropy of attributes given object classes to encode the relevance, i.e. we define *relevance*$(A_i|C_j) = 1/H(A_i|C_j)$ as computed on normal objects. The intuition behind this is that if the distribution $P(A_i|C_j, N)$ is peaky,

it should be relevant and discriminative for the purpose of measuring normality/abnormality, while a uniform distribution is not really useful.

Lastly the performance of attribute classifiers is not consistent across different attributes; some attributes are harder to learn than others. Attribute classifiers do not work perfectly even on normal images, which can result in unreliable measures that can affect inference about abnormality. To measure attribute reliability, we compute the accuracy of attribute classifiers evaluated on a validation set. A measure of reliability can be defined as $\textit{reliability}(A_i) = acc(A_i)$, where $acc(A_i)$ is the accuracy of the classifier for attribute A_i, which ranges between 0.5 and 1. Now we can a define relevance-adjusted accuracy-adjusted surprise measure of observing attribute classifier response a for attribute A_i and class C_j as

$$\textit{surprise}_{(A_i|C_j)}(a) = \textit{reliability}(A_i) * I(A_i = a|C_j) * \textit{relevance}(A_i|C_j) \tag{3.1}$$

The surprise is a function $\textit{surprise}_{(A_i|C_j)} : [0,1] \rightarrow [0,\infty)$ that is defined for each pair of categories and attributes, which takes the output of attribute classifier and assesses the strangeness/abnormality in that score.

Notice that, to learn the model we only need normal images. The relevance factor, based on the conditional entropies, is computed during the training time on normal images and will appear as a fixed term for each combination of attributes and object classes. The reliability factor is only measured offline on normal training images.

3.2.3 Attributes Responsible for Abnormalities

Each abnormality prediction for an image can be supported by a set of abnormality causes in terms of attributes. The surprise measure in Eq. 3.1 directly gives us a measurement of how one given attribute might be the cause of abnormality. However, there are two possible reasons that can cause a given attribute to be surprising: either the attribute is typical within the class and is missing in the observed image, or the attribute is not typical for the object class and exists in the image. Both cases will results in low attribute likelihood given the category and therefore, high surprise value. It is useful to discriminate between these two cases for the purpose of abnormal attribute reporting. To achieve this we define a signed surprise function

Table 3.1: Evaluation of different approaches for categorizing abnormal images. Percentage accuracy is shown.

Task	Method	Features	test dataset	Airplane	Boat	Car	Chair	Motorbike	Sofa
Categorization	one-vs-all SVM	base	PASCAL	81.83	74.67	76.67	81.0	81.5	81.5
Categorization	one-vs-all SVM	base	Abnormal	55.92	75.69	68.23	72.77	64.67	46.03
Categorization	one-vs-all SVM	Attributes	PASCAL	78.66	61.17	63.33	65.33	77	82.17
Categorization	one-vs-all SVM	Attributes	Abnormal	58.99	64.67	70.65	73.09	73.58	64.18
Categorization	one-class SVM	Attributes	PASCAL	76.85	77.45	76.55	77.59	75.09	76.37
Categorization	one-class SVM	Attributes	Abnormal	71.05	69.50	59.90	67.99	67.28	63.65
Detection	Part-based	HoG	Abnormal	5 %	3 %	35 %	0 %	10 %	0 %

$\textit{signed\_surprise}_{(A_i|C_j)} : [0,1] \rightarrow (-\infty, \infty)$ as

$$\textit{signed\_surprise}_{(A_i|C_j)}(a) = \textit{surprise}_{(A_i|C_j)}(a) * (2 * a - 1). \tag{3.2}$$

This function encodes absence of expected attributes and presence of unexpected attributes by projecting scores to the range $-\infty$ to $+\infty$ respectively . This score takes into account the probability of being a normal attribute and attribute classifier response.

3.2.4 Features and Attributes

We describe and model objects using visual attributes, which can be categorized into shape, color, texture and part related attributes. To learn a broad range of attributes we need a wide variety of features, which we call "base features". Similar to [53, 52] we use edges to model the shape, and pyramid of Histogram of Oriented Gradient (HoG) features to find part attributes. ColorSIFT and Texture features are extracted to learn attributes that are related to material and texture. Base feature extraction has been done in a pyramid-based approach. we divide the image into six patches and extract base features for each of these patches in addition to the whole image. We apply canny edge detector, quantized output of HoG and Texton filter bank responses. Also, unlike[53, 52] we use ColorSIFT to improve features for learning attributes related to color and material. This feature extraction process will result in a 10751-dimensional feature vector for each image.

We use 64 visual attributes, where each of them is modeled via Support Vector Machine (SVM) classifier that uses selected dimensions of base feature vectors. In order to find out

which dimensions of base feature vectors are important for a specific attributes, we fit a l_1-regularized logistic regression between objects coming from a specific class with that attribute and without it.

3.2.5 Evaluation of Object Recognition Models on Abnormal Objects

To investigate the performance of the state-of-the-art algorithms, when applied on abnormal images, we performed several evaluation experiments. This evaluation is also fundamental to our approach since we use the categorization result as the first stage in our approach.

- *Detection:* In this case we evaluated the detection performance not categorization. For each image in our abnormal dataset we ran a detector based on each category and evaluated the detection result, i.e., for car images we ran a car detector. We hypothesize that this approach should fail when applied on abnormal images because abnormal images do not exhibit normal part configuration We use the state-of-the-art deformable part-based detectors of [58] to evaluate how well one can categories images of abnormal objects. Here, we do not compute the localization based performance measures. Therefore we relax the overlap constrain to zero. This means that we want to use this detector as a classifier and it would be a correct response if the detector fires on an image that contains instances of desired category. We hypothesize that this approach should fail when applied on abnormal images because abnormal images do not exhibit normal part configuration. Numbers in Table 3.1 shows the percentage of the cases where the detector could do classification correctly.

- *Categorization - Base features:* Each image is represented using base features and one-vs-all SVM classifiers are trained for each category.

- *Categorization - Attribute based classifier [53] for categorization:* Each image is represented by a feature vector which is the output of 64 attribute classifiers. We trained two different classifiers: one-class SVM classifier for each category and a one-vs-all SVM classifier. The one-class SVM only has access to positive examples of each class during training.

In all cases the models were trained on subsets of PASCAL images (denoted as the normal dataset) and no training is done on the abnormal dataset. For part-based detectors we used the trained models provided by Felzenszwalb et al. [58] (also trained on PASCAL). We evaluated on both the normal (600 images from PASCAL test) and our abnormal dataset. The results are

shown in Table 3.1. It is surprising to see the large divergence in the results, while part-based detectors failed, as expected to detect the objects, the attribute-based and the base-features categorization approaches is consistently able to categorize the abnormal images. Of course the performance on categorizing abnormal images is not as good as the case of normal images in most of the cases, which is expected, but the generalization to the unseen abnormal images is quite surprising. There are even cases where the performance on abnormal images is better than the normal test images.

There are various conclusions and observations we can make out of this experiment. *First*, we can reject the hypothesis that the bad performance for part-based detectors is because of different biases in the abnormality dataset, since the categorization approaches performed consistently on it. *Second*, failure of part-based detectors might be used as a strong cue of abnormality in an image given that we actually have another way to detect the object and correctly categorize it! *Third*, it is clear that the attribute-based approach captures a good representation of each category that carried over for unseen test instances from both the normal and abnormal datasets.

3.2.6 Evaluation of Preliminary Models for Abnormality Classification

We evaluated the various proposed methods for modeling typicality given the category as described in Sec 3.2. For all these experiments we trained the typicality models using the same training data from PASCAL train set. The number of images per class varies as indicated under the category name in Table 3.2. For testing we used a mixture of normal images from PASCAL (100 per class) and abnormal images from our dataset (100 per class). Since the goal is to evaluate the Normality/Abnormality classifiers given the class, out of these test images we only used the ones that are correctly categorized by the first stage categorization. The baseline for this experiment is a typicality model learned on the result of the first stage categorization classifier. We used the confidences from the one-vs-all SVMs used for categorization and fit a Gaussian model for the distribution of the confidences for the typical images of each class. We use this Gaussian Model to obtain a probability of being typical given the category.

The performance of the Normality/Abnormality classifiers for each category is measures via Area Under the Curve (AUC) as reported in Table 3.2. On average the Naive Bayes approach

Table 3.2: Normality/Abnormality Classification Results

Normality/Abnormality Classification within each category (AUC)

Approach \ Object class	Airplane 270	Boat 353	Car 922	Chair 811	Motorbike 197	Sofa 153	Average
Baseline	0.5183	0.7397	0.5671	0.9211	0.6682	0.6011	0.5597
Naive Bayes	0.6230	0.9394	0.8847	0.9882	0.8136	0.7149	0.8273
Naive Bayes with Attribute relevance	0.6638	**0.9403**	0.9166	0.9876	0.8021	0.6919	**0.8337**
Nonparametric Model	**0.7265**	0.7917	0.6629	0.5057	**0.8681**	**0.7963**	0.7252
Global Manifold Distance	0.5280	0.8887	**0.9318**	**0.9901**	0.7437	0.6429	0.7875
Local Manifold Distance	0.5771	0.7480	0.7376	0.9404	0.7437	0.7196	0.7444
Manifold-based Density Model	0.6406	0.8196	0.7990	0.9218	0.7009	0.6238	0.7510
One class SVM	0.6615	0.9370	0.9222	**0.9901**	0.8140	0.6693	0.8324

Table 3.3: Evaluation of abnormal attribute reporting - KL divergence from ground truth annotation generated by human subjects in Turk experiment.

Evaluation of abnormal attribute reporting - KL divergence from ground truth

Approach \ Object class	Airplane	Boat	Car	Chair	Motorbike	Sofa	Average
Baseline(1)	0.0796	0.08	0.0775	0.1035	0.0944	0.064	0.0832
Baseline(2)	0.0826	0.0768	0.0809	0.0956	0.0892	0.0565	0.0803
Our Approach	0.05669	0.03689	0.07583	0.06315	0.06349	0.06954	0.0609

with attribute relevance gives the best results, with almost similar result using the one-class SVM. The global manifold distance model gives the best results for the Car and Chair categories where there are a lot of training samples, while it does not perform as well for the categories with small number of samples. This is expected since any manifold approach needs a dense sampling of the underlying manifold. We hypothesize that the manifold model should give the best results if all categories have enough training data.

3.2.7 Abnormality Prediction via Probabilistic Models

The task of abnormality prediction is to label images in the test set as either normal or abnormal. Given an attribute vector for each image, our approach will assign a probability of being normal. The complement of this probability can be used as an abnormality score, denoted as "*Graphical model*" (please see Section 3.2). We also use the surprise scores explained in section 3.2.2 for the enhanced model, which we call "*Graphical Model with surprise score*". In that model the

Method	AUC
One class SVM (learned on Normal)	0.5980
Two class SVM (leaned on Abnormal and Normal)	0.8657
Graphical Model for abnormality prediction	0.8703
Graphical Model with adjusted surprise scores	0.9105

Table 3.4: Evaluation of Abnormal Detection approaches (AUC)

surprise score is used to compute a robust version of $P(A_i|C_j, N)$, taking the relevance and reliability of attribute into consideration. We learn the models for $P(A_i|C_j, N)$, *relevance*, and *reliability* measures only from normal images. *Relevance* term in the graphical model is directly related to conditional entropy of attribute values given object class. For a combination of (class,attribute) we compute Shannon conditional entropy using normal images in PASCAL dataset. These entropies are computed for all possible combinations of attributes and object class once during the training time. For the *Reliability* of an attribute, we tested its classifier on PASCAL test dataset and normalize its accuracy to the interval (0,1). This will result in one *Reliability* score for each individual attribute. In the test time, $P(A_i|C_j, N)$ is computed by evaluating learned distributions at attribute responses for each test image. By aggregating these probabilities over all combination of attributes and classes for an image, we get the probability of being normal.

We compare our abnormality prediction with that of one-class SVM, which is widely used for abnormality prediction [31]. We train a one-class SVM using attributes of positive examples from each object classes (in the normal image dataset). We used the confidence of these one-class SVM as scores of normality and measured its accuracy for abnormality prediction by AUC (normal vs abnormal classification).

The results of these probability based models for the Normality/Abnormality prediction in images are shown in Table 3.4. We use AUC to measure how well each method performs. Our method not only outperforms the baseline(one-class SVM), but also outperforms all the preliminary model(Non-probabilistic models of Table 3.2) . Adding the relevance term and attribute classifier reliability improves our original model.

We also compared our method with an abnormality classifier trained on both normal and

abnormal images. For this classifier(second row in table 1), we learn a two class SVM on top of visual attributes to learn a boundary between normal and abnormal images. Normal images are selected from PASCAL train dataset and equal number of abnormal images have been chosen from abnormal dataset. Our model, without observing any instance of abnormal images, outperforms this baseline that is learned on both abnormal and normal images.

Abnormal images are not equal in terms of how strange they look like to human. This has been shown in the human subject experiment when each image gets different votes for being abnormal. Our abnormality score can also impose a ranking on abnormal images. Figure 3.5 shows ranked abnormal images for cars and boats. From left to right the abnormality of images increases.

3.2.8 Abnormal Attribute Reporting

After detecting an image as abnormal, we recognize its abnormality causes in terms of visual attributes. Our proposed graphical model assigns a surprise score for each attribute in an abnormal image. We used the same training and testing setting as above. In the first step, we predict top categories for each abnormal image as its object class. As we discussed in section 3.2.3, assuming an image belongs to a specific class, each attribute will have a surprise factor. Abnormal attributes have extreme values as their surprise factor with a negative sign for missing attributes and positive sign for unexpected ones. Figure 3.4 shows some abnormal images and their corresponding output of our model for the task of abnormal attribute reporting. Here we report first two candidates for object class and their corresponding *Missing (M) attribute* or *Unexpected (U) attribute*.

We use ground truth rating from the MTurk responses to quantitatively evaluate our abnormal attribute reporting. As we explained in section 2.2.1 each abnormal image in our dataset, has a user score for four different causes of abnormality (Shape, Color, Texture and Pose). Since our model evaluates strangeness of attributes individually for an image, we grouped the attributes together based on their relatedness to each of these four cases. With this grouping, we can aggregate and normalize the scores for each abnormality cause. These surprising scores for each category of attributes can be compared to those we have in MTurk annotation. Table 3.3 reports Kullback-Leibler divergence between distribution of surprising scores for each

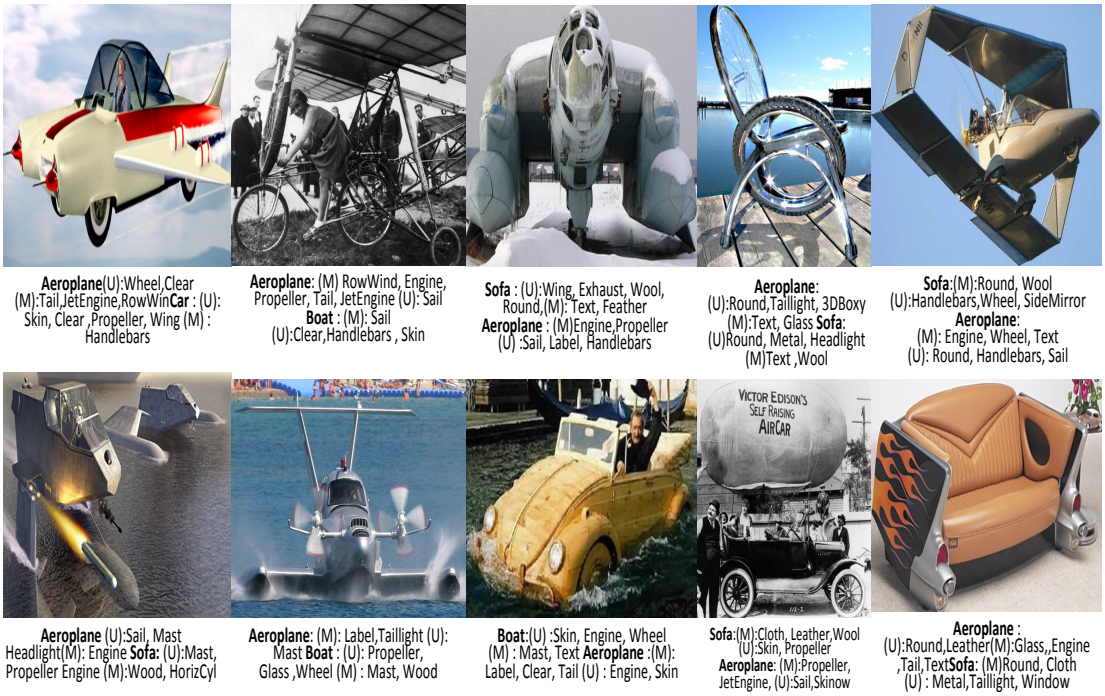

Figure 3.4: Abnormal image describing: Class prediction, Missing(M) and (U)Unexpected attribute reporting

SVM classification before abnormality detection	47.2502
SVM classification after abnormality detection	38.5203

Table 3.5: Evaluation of abnormal object categorization - KL divergence from ground truth

abnormality cause made by our approach and the ground truth MTurk annotation.

In this experiment baselines are based on Farhadi et al[53]. First row of Table 3.3 is regarding to the experiment when we take the mean and variance of each attribute for all the training data for a specific class. We get a 64 dimensional attribute mean μ_i and variance σ_i^2 for each class i. For each test image after categorization and abnormality detection, if the image is an abnormal instance of class j then we will focus on detecting abnormal attributes. Attribute$i$ in test image will be abnormal for class j if the attribute confidence does not fall in the range of $2 * \sigma_i$ around the attribute mean for class$j$. Baseline (2) is similar to the previous experiment, but this time we increased the range interval to $4 * \sigma_i$ We evaluated the abnormality reporting using our approach as described in section 3.2.3.

Figure 3.5: Abnormal image ranking. Abnormality score increases as we move to the right side

3.2.9 Categorization of Abnormal Objects

We are interested to evaluate the performance of object classifiers, when they are provided with additional information about the abnormality of the object. Abnormal image categorization is a subjective task; there might not be one correct answer. Therefore, we use the responses of MTurk users to generate distribution over categories for each images. Our model can also produce such a distribution by assigning a class confidence out of 6-way SVM classifier to each image. We compare the KL-divergence between our model and human generated distribution as a way to measure the performance of our classifier.

Interestingly, attribute-based classification of objects can be improved by reasoning about abnormality. Knowing that an object is abnormal along with the list of attributes that cause the abnormality should help categorizing that object. The normal category models are trained on the attributes of normal images. By discounting the abnormal attributes in category models, one can improve the categorization of abnormal images. More specifically we train a linear classifier for each category of normal objects in the attribute space, and by controlling the influence of the dimensions corresponding to problematic attributes we can discount the effects of abnormalities. We do this by replacing the current value of problematic attributes with its average value conditioned on classes.

We re-run the same SVM classifier on abnormal images, but this time the effect of abnormal attributes for classification has been adjusted. Second row of Table 3.5 shows that by this refinement the distribution over different object classes for abnormal images gets more similar

to what people have guessed about it. This has been indicated by a lower KL divergence number for the second row in Table 3.5 comparing to its first row. Last row in Table 3.5 refers to the case that each class has a surprising score given a set of attribute responses in an image, inverse of these surprising factors for each object category shows the class-membership confidence.

Chapter 4

Computational Models for Abnormality Recognition

In this section, we propose a computational model to find abnormal images and reason about them based on three scores that come from three major reasons of abnormality. We start by investigating normal images and proposing a model for relating elements of an image: object, context and scene. Next, we derive a set of scores, called "Surprise Scores", to measure how abnormal an image is with respect to these elements. Later we explain how we merge the different scores to decide if the image is abnormal or not, and finally find the dominating abnormality reason that affects this decision.

4.1 Modeling Typicality

We propose a Bayesian generative model for typical scenes and objects, depicted in Figure 4.1. This model formulates the relation between objects, context and other information in the scene that is not captured by objects or the context (e.g. scene characteristics such as Sunny or Crowded). This is a model of typicality, and atypicality/abnormality is detected as a deviation from typicality. Hence, this model is trained using only typical images and relies on visual attributes and categories of both objects and scenes.

Visual attributes have been studied extensively in recognition [92, 120]. In contrast to low-level visual features (e.g. HOG, SIFT), attributes represent a valuable intermediate semantic representation of images that are human understandable (nameable). Example attributes can be "Open area", "Sunny weather" for scenes and "wooden" or "spotty" for objects. Attributes are powerful tools for judging about abnormality. For example, the object-centric model of [146] mainly used attribute classifiers to reason about abnormality. However, the response of an attribute classifier is noisy and uncertain. As a result, we categorize the object based on low-level visual features apart from its attributes scores. Later, out model at the level of the object

focuses on deviations between categories of the objects and its meaningful visual characteristics (attributes). In short, if low-level features predict an object to be a car, while attribute responses do not provide evidence for a car, that is an indication of abnormality.

As a similar argument stands at the level of scenes, we model the typicality of low-level visual features (F) and attributes (A) for both objects (O) and scenes (S). Figure 4.1 shows that assuming we observe a normal image *I*, any distribution over scene category S imposes a distribution over the categories of objects O that are present. This procedure holds for all K objects in the image (left plate is repeated K times). Each object category imposes a distribution over object's low-level features F^o and attributes A^o. Similarly, scene categories impose a distribution over scene's low-level features F^s and attributes A^s. However, extracted visual features for scenes are different from ones extracted for objects. We define two disjoint sets of attributes for objects ($A^o = \{A_i^o\}_1^n$) and attributes for scenes ($A^s = \{A_i^s\}_1^m$).

Learning the model involves learning the conditional distribution of object-attribute, given object categories ($\{P(A_i^o|O_k), i = 1 \cdots n, k = 1 \cdots V\}$), and scene-attribute conditional probability distribution given scene categories ($\{P(A_i^s|S_j), i = 1 \cdots m, j = 1 \cdots J\}$), where each of these distributions is modeled as a Gaussian. We also learn probabilities of object categories given scene categories ($\{P(O_k|S_j), k = 1 \cdots V, j = 1 \cdots J\}$) , where V and J are number of object and scene categories.

4.2 Measuring Abnormality of Images

For a given image, we measure how abnormal it looks like based on three three surprise scores. These scores are inspired by the taxonomy that we learn by analyzing human responses in Chapter 2. In following sections, we describe each one of these surprise scores: Object-centric, Context-centric, and Scene-centric.

4.2.1 Scene-centric Abnormality Score:

For any scene category, some visual attributes are more relevant (expected). This is what we call relevance of i^{th} scene attribute for the j^{th} scene category, denoted by $\Omega(A_i^s, S_j)$ [1]. We compute this term by calculating the reciprocal of the entropy of the scene-level attributes for a given scene category $\Omega(A_i^s, S_j) = 1/H(A_i^s|S_j)$ over normal images. This relevance term does not depend on the test image.

For a given image, applying scene classifiers produce a distribution over scene categories. Assuming a scene category, we compute the information content in each scene-attribute classifier response ($I(A_i^s|S_j) = -logP(A_i^s|S_j)$). This information content is a measure of the surprise by observing an attribute for a given scene class. Since attribute classifiers are noisy, depending on the concept that they are modeling, we need to attenuate the surprise score of a given attribute by how accurate is the attribute classifier. We denote this term by $\Upsilon(A_i^s)$, which measures the accuracy of the i^{th} scene attribute classifier on normal images. Therefore the scene surprise score (*Surprise*$_S$) is computed by taking the expectation given $P(S_j)$ as following:

$$\sum_j P(S_j)[\sum_i I(A_i^s|S_j)\Upsilon(A_i^s)\Omega(A_i^s, S_j)] \tag{4.1}$$

4.2.2 Context-centric Abnormality Score:

An image looks abnormal due to its atypical context if one of the following happens: first, an unexpected occurrence of object(s) in a given scene. (e.g. elephant in the room); second, strange locations of objects in the scene (e.g. a car on top of the house); or inappropriate relative size of the object. We propose Eq. 4.2 to measure the context-centric surprise (Surprise$_C$) of an image based on aforementioned reasons:

$$\sum_k \sum_j \Lambda(O_k)[\hat{I}(O_k|S_j) + I(L_k|O_k)]. \tag{4.2}$$

[1] For simplicity, we slightly abuse the notation and use A_i^s to denote both the i^{th} attribute, and the i^{th} attribute classifier response for scene attributes. The same holds for object attributes as well.

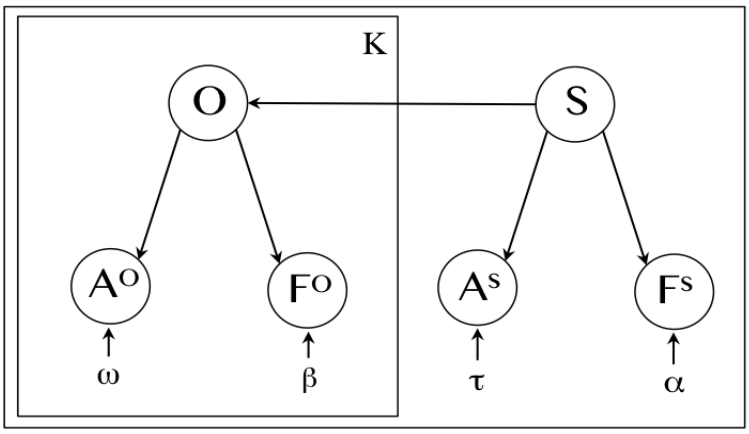

Figure 4.1: Each image relates to one scene (S) and represents K objects. Categories are detected via visual features (F), and described by visual attributes (A).

The term $\hat{I}(O, S)$ measures the amount of surprise stemming form the co-occurrence of the objects in the scene (Eq. 4.3). We measure the surprise associated with each object classes appearing in the scene S_j by computing the information content of each combination of scene categories and object classes, $I(O_k|S_j)$, modulated by the probability of the object and scene categories.

$$\hat{I}(O_k|S_j) = P(S_j)P(O_k)I(O_k|S_j). \tag{4.3}$$

On the grounds that we use a distribution as the output of classifiers rather than a single class confidence, we do not need to involve the accuracy of neither the object classifier nor the scene classifier to tackle the uncertainty output.

The term $I(L_k|O_k)$ measures how much surprising is the location of the object k in the image. Assuming we know category of the object (O_k), we expect to see it in certain locations in the image. By considering one object category at a time, we learn a distribution of possible locations for the object in normal images and use it to compute the information content of the object location in a test image.

Finally we aggregate the co-occurrence and location term and modulate the score by multiplying it with $\Lambda(O_k)$, which stands for the importance of the size of the object relative to the whole image in judging the context atypicality. If the object of interest is tiny or huge in the image, the contextual surprise should be modulated down. To model $\Lambda(O)$ for each object

category(O) we learn the distribution of its relative size by considering the normal images with typical context and for the test image compute its probability based on this distribution.

4.2.3 Object-centric Abnormality:

For *Surprise*$_O$ we check if the objects in the image look typical or not independently. We assume that we take the object out of the scene and measure how abnormal it is based on its predicted object class and visual attributes. This term is in part similar to work of Saleh et al [146]. However, we are different from their work as we classify the objects based on low-level visual features F^o rather than visual attributes A^o. We formulate the object-centric surprise score (Surprise$_O$) as:

$$\sum_k P(O_k) * (\sum_i I(A_i^o|O_k) * \Upsilon(A_i^o) * \Omega(A_i^o, O_k)) \tag{4.4}$$

Where $P(O_k)$ is the distribution over object categories obtained from low-level visual features. $I(A_i^o|O_k) = -log(P(A_i^o|O_k))$ denotes the amount of the surprise by observing the response of the i-th attribute classifier, given class O_k. Similar to scene-centric surprise score, $\Upsilon(A_i^o)$ adjusts the weights of visual attributes based on how reliable one attribute performs on normal images. $\Omega(A_i^o, O_k)$ models the relevance of attribute A_i^o to object k, however this is computed based on ground truth annotation rather than the conditional entropy of attributes.

4.2.4 Parametric Model for Typicality

For the final decision about abnormality of an image we should compare the three surprise scores and pick the maximum as the the most important reason of abnormality. However, there are two issues that prevent us from using the maximum of raw surprise scores. These described surprise scores are based on quantifying the information content, therefore these measures are unbounded (as the probability approaches zero, the surprise approaches infinity). The other issue is that these surprise scores are not comparable since the information content in each of them are modulated differently. As a result it is hard to compare the values of *Surprise*$_O$, *Surprise*$_S$, and *Surprise*$_C$ to determine which of these reasons gives rise to the abnormality in the image, if any. To tackle these issues, we propose to model the distribution of the surprise

scores for normal images.

Toward this goal, we compare fitting different parametric models to the empirical distributions of three surprise scores, computed over normal images. For model selection we consider simplicity of the distribution, as well as how well it fits the empirical data based on Akaike Information Criterion (AIC) [2]. We are interested in simpler distributions, because of their better generalization and the ability to derive simpler (sometime closed from) CDFs. Our experiments show that independent of the reason of abnormality, surprise scores follow exponential family of distributions. We pick *"Inverse Gaussian"* distribution as the underlying distribution. Due to limited space, we put more analysis in the supplementary material. Given these probabilistic models, we can compute the probability of observing a given surprise score instead of the raw surprise scores. Then we can classify the reason of abnormality in an image by comparing the CDFs of the parametric models, i.e.,

$$\underset{o,s,c}{\operatorname{argmax}}(\phi_o(Surprise_O), \phi_s(Surprise_S), \phi_c(Surprise_C)) \tag{4.5}$$

Where $\phi_o(\cdot)$,$\phi_s(\cdot)$,$\phi_c(\cdot)$ are the inverse Gaussian CDFs for the object, scene, and context - centric parametric surprise models respectively. Parameters of each model are estimated only from the normal training data.

4.3 Experimental Results

4.3.1 Object-centric Typicality Modeling

We train our model for abnormality prediction on six classes of objects: Airplane, Boat, Car, Chair, Motorbike and Sofa. We choose these categories to be comparable with related work [146]. Based on our experiments state-of-the-art object detectors [66, 49] generally fail to detect abnormal objects. As a result, we assume that object bounding boxes are given in the image. Through our experiments, we convert confidences of classifiers (e.g. attribute classifiers) to the probability by using Platt's method [126].

Reason-name	Var. I	Var. II	Var. III	Full-score
Object-centric	0.6128	0.7985	0.8050	0.8662
Context-centric	0.6778	0.6923	0.8255	0.8517
Scene-centric	0.6625	0.7133	0.7210	0.7418

Table 4.1: Ablation experiment to evaluate the importance of different elements of each surprise score (rows) for the task of abnormality classification (Area Under Curve - AUC). For scene-centric and object-centeric: Var.I) Only $I(A|S(orC))$, Var.II) Full-score without relevance ($\Omega(A, S(orC))$), Var.III) Full score without attribute accuracy ($\Upsilon(A)$). For context-centric: Var.I) $I(O|S)$, Var.II) $I(O|S) * \Lambda(O_k)$, Var.III) $I(O|S) + I(L|O)$.

Object Classification

We use "Kernel Descriptors" of Bo et al [19] to extract low-level visual features for each object. We specifically use *Gradient Match Kernels, Color Match Kernel, Local Binary Pattern Match* kernels. We compute these kernel descriptors on fixed size 16 x 16 local image patches, sampled densely over a grid with step size 8 in a spatial pyramid setting with four layers. This results in a 4000 dimensional feature vector. We train a set of one-vs-all SVM classifiers for each object class using normal images in PASCAL train set. We perform five-fold cross validation to find the best values for parameters of the SVM. This achieves in 87.46% average precision for the task of object classification in PASCAL2010 test set. Object classification in abnormal images is extremely challenging and state-of-the-art approaches cannot generalize to atypical objects (see Table 5.1). Our learned object classifiers achieve top-1 error 67.25% on abnormal objects.

Object Attributes

We use the annotation of 64 visual attributes for the objects in "aPASCAL" dataset [53]. Farhadi et al [53] extracted HOG, color, edges and texture as base features and learned important dimensions of this feature vector for each attribute using $l1$-regularized regression. However, we do not extract edges and we extract colorSIFT [166] rather than simple color descriptors. Also we do not perform the feature selection and use the original base features. Our approach for learning attribute classifiers outperform pre-trained classifiers of [53] for the task of attribute prediction on aPascal test set.

Experiment Number	Method	Accuracy	Training images		Testing images	
			Normal	Abnormal	Normal	Abnormal
I	Object-centric baseline [146]	0.9125	Pascal	Not Used	Pascal	Dataset of [146]
	Our Model - Object-centric	**0.9311**	Pascal	Not Used	Pascal	Dataset of [146]
II	Context-centric baseline [122]	0.8518	SUN	Not Used	SUN	Subset of [122]-without human
	Our Model - Context-centric	**0.8943**	Pascal	Not Used	SUN	Subset of [122]-without human
III	One Class SVM - based on Attributes	0.5361	Pascal	Not Used	Pascal	Our dataset
	Two Class SVM - based on Attributes	0.7855	Pascal	Our dataset	Pascal	Our dataset
	One class SVM - based on Deep features (fc6)	0.5969	Pascal	Not Used	Pascal	Our dataset
	Two class SVM - based on Deep features (fc6)	0.8524	Pascal	Our dataset	Pascal	Our dataset
IV	Our Model - No Object-centric score	0.8004	Pascal	Not Used	Pascal	Our dataset
	Our Model - No Context-centric score	0.8863	Pascal	Not Used	Pascal	Our dataset
	Our Model - No Scene-centric score	0.8635	Pascal	Not Used	Pascal	Our dataset
	Our Model - All three reasons	**0.8914**	Pascal	Not Used	Pascal	Our dataset

Table 4.2: Evaluating the performance (AUC) of different methods for classifying normal images vs. abnormal images.

4.3.2 Context-centric Typicality Modeling

Following Eq. 4.2 we compute the amount of information provided by the co-occurrence, location and size of the objects in the scene. For modeling the co-occurrence we use the annotation of SUN dataset and learn the conditional entropy of object categories for each scene category. To learn the typical location of objects in images and their relative size, we use PASCAL context dataset [107] that annotated PASCAL images with semantic segmentation. For this purpose we divide PASCAL images into equally-sized grids and for each grid compute the probability of the number of pixels that belongs to each object category. We learn these distributions over all images that are labeled as positive samples of the object category. Our experiments show that the ratio of pixels that contribute to a specific object in a grid, follows an *Exponential distribution*. We model the normal relative size (the ratio of object to the whole image) with a *Gamma distribution*.

4.3.3 Scene-centric Typicality Modeling

To model the typical scene and context, we use the annotation of SUN dataset [174] to find most frequent scene categories for our six object classes. We start with top ten scene categories for each object class and merge them based on similarities in images, which results in 4700 images of 16 scene categories. . For example, we merge Airfield, Airport, Runway and Taxiway into one category.

Scene Classification

State-of-the-art for the task of scene classification [84, 41, 185] use image collections and scene categories that are different from our experimental setting. As a result, we train scene classifiers specifically for our experiments by following the approach of Parizi et al [121]. However, we modify the process of selecting image patches during training classifiers. This approach outperforms prior arts for the task of scene categorization of normal images in our collection by achieving 94% average precision over 16 scene categories in our train set.

Scene Attributes

We use 102 scene-level visual attributes proposed by Patterson et al [123]. We follow the strategy of [123] to train attribute classifiers using images of normal scene. We measure the attribute reliability $\Upsilon(A_i^S)$ and relevance of an attribute for a scene category, in terms of the conditional entropy of the attribute confidences of all normal images from the same scene category: $H(A_i|S_j)$. We also estimate the conditional distribution of attribute responses in normal images for a given scene category, as a normal distribution and later use this probability in computing $I(A_i|S_j)$ for abnormal images.

4.3.4 Abnormality Classification and Reasoning

We compute all three ***Object-centric, Context-centric*** and ***Scene-centric*** surprise scores following Eqs. 4.1,4.3 & 4.4. We use these surprise scores to first, classify an image as abnormal vs. normal (abnormality classification). Next, we use the parametric model for abnormality classification and finding the reason of abnormality that contributes the most to our final decision (abnormality reasoning). In the first step, we conduct an ablation experiment to evaluate the performance of each surprise score, and its components for distinguishing normal vs. abnormal images. Table 4.1 shows the result (AUC) of this experiment, where each row represents a specific reason and columns are different variations of the corresponding surprise score. In each row, we consider the abnormal images of that specific reason as the positive set and all normal images along with other abnormal images (due to a different reason) as the negative set.

Table 4.1 shows that for all reasons of abnormality, the full version of surprise scores –

Figure 4.2: Ranking of abnormal images of cars based on different reasons of abnormality

all components included – achieves the best result (last column). For object and scene-centric surprise scores, Var. I represents a variation of the surprise score, which only uses the term $I(A|S(orC))$. We can improve this basic score by adding "the accuracy of attribute classifiers" (in Var. II), or "relevance of the attribute to the object/scene category"(in Var. III). We conclude that both components of relevance and attribute accuracy are equally important for improving the performance of abnormality classification. For context-centric surprise scores, the location of the object (conditioned on its category) is by a large margin, the most important factor to improve the basic surprise score (in Var. I) – which only finds the irregular co-occurrence of objects and scene.

These reason-specific surprise scores can be used for sorting images based on how abnormal they look like. Figure 4.2 shows some examples of these rankings for images of cars. Each row corresponds to one reason of abnormality, where images of abnormal cars are selected from our dataset and sorted based on the corresponding surprise score. Supplementary material includes more images of ranking experiment, histograms of these individual surprise scores for normal vs. abnormal images and the corresponding fitted probability functions.

We compute the final surprise score of an image based on the Eq. 4.5, where we use the index of maximum surprise score for the task of abnormality reasoning. Table 4.2 shows the performance (AUC) of our final model for the task of abnormality classification in four different experiments (four boxes), where the last four columns indicate the source of images that we use for training and testing. Comparing the first two rows show that we outperform the baseline of object-centric abnormality classification [146] on their proposed dataset. This is because we learn better attribute classifiers and compute the surprise score by considering all possible categories for objects. Box II in table 4.2 shows that our proposed context-centric surprise score outperforms state-of-the-art [122] for contextual abnormality classification. It should be mentioned that Park et al [122] originally performed the task of abnormal object detection. For the sake of a fair comparison, we change their evaluation methodology to measure their performance for the task of abnormality classification.

Box III in Table 4.2 shows the results of another baseline experiment for abnormality classification, where all abnormal images are used at the test time (despite box I& II). We train one-class (fifth row) or two-class (sixth row) Support Vector Machines (SVM) classifiers, where the later case performs better. Although we do not use abnormal images in training, our model still outperforms the two-way SVM classifier that is trained via both normal and abnormal images. This is mainly due to the fact that abnormality is a graded phenomena and a generative model finds abnormality better than discriminative ones. To evaluate the importance of each reason-specific surprise score in the parametric model, we conduct an ablation experiment as it is reported in box IV of the Table 4.2. In each row, we remove one reason of abnormality and compute the parametric model based on the other two surprise scores. Comparing these performances with the one of full model (last row) show that object-centric surprise score is the most important element of the final model, as removing it results in the biggest drop in the performance. Also the context-centric seems to be the least important reason for detecting abnormal images.

We use three reason-specific scores of abnormality to visualize abnormal images in a 3-D perspective. Figure 4.3 shows this plot, where axis are surprise scores and data points are images, color coded based on the main reason of abnormality. For example, red dots are images that the most dominant reason of abnormality for them is object-centric. In this plot, we

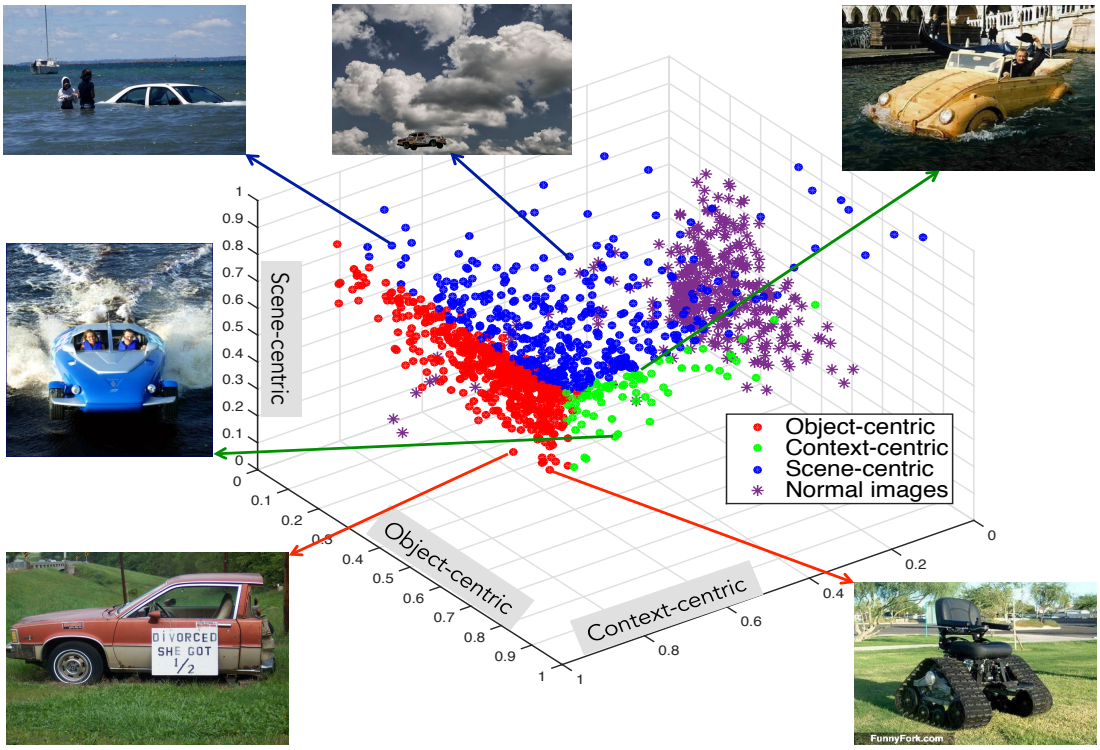

Figure 4.3: Plotting images in 3-D based on three surprise score that we get from our computational models. Points are colored based on the most important reason of abnormality in each image. Colorful clouds of abnormal images are separated from normal images (spread of purple stars close to the center of coordinate).

see how continuous surprise scores can spread abnormal images and we can find some boundaries between the main reasons of abnormality (three axes). More importantly, normal images (purple stars) are separable from abnormal images using these surprise scores.

In order to evaluate the quality of our model for reason prediction (abnormality reasoning), we compute the KL-divergence between three surprise scores of our model and the ground truth surprise scores for each image. We compute the ground truth scores by grouping and taking the average of response for 21 fine-grained reasons of abnormality in Turk experiment. We group these 21 fine-grained reasons based on the adopted taxonomy, and aggregate the corresponding responses to get three main surprise scores. We take the average of theses scores over all annotators for one image. The We measure the KL-divergence between scores of our final model and ground truth scores as 0.8142. Average human annotator predicts scores with KL-divergence of 0.6117. Interestingly, if we only use three raw surprise scores as the predicted scores, KL-divergence increases to 2.276. This verifies the value of parametric model

Object-centric abnormality	301	62	41
Context-centric abnormality	230	88	43
Scene-centric abnormality	96	24	105

Table 4.3: Confusion matrix for the task of abnormality reasoning. Rows are predicted labels and columns are ground truth given by the majority vote in the Turk experiment.

for predicting more meaningful surprise scores, which are more similar to human judgments.

In the last experiment, we classify abnormal images into three reasons of abnormality (abnormality reasoning) by picking the index of the reason that gives the highest surprise score. We compute the confusion matrix for this prediction as it is shown in Table 4.3, where columns are ground truth labels and rows are the predicted labels.

Chapter 5

Typicality Estimation for Learning Better Object Classifiers

5.1 Introduction

Convolutional Neural Networks (CNN) have made remarkable progress in a variety of computer vision tasks. To just name few of the recent advances, CNN-based models greatly improved object classification and detection [152], image retrieval [149], scene classification [184], and image captioning [168].

Despite the superior performance on large-scale visual object classification, convolution neural networks cannot emulate the generalization power of the human visual system in real-world object categorization [65, 125], especially when it comes to objects that differ substantially from the training examples. Figure 5.1 shows examples of these atypical images, which human subjects categorize correctly, but which a CNN model misclassified with a high confidence. We evaluate the performance of CNNs for the purpose of object classification on atypical images. Humans are capable of perceiving atypical objects and reasoning about them, even though they had not seen them before [146]. But our experiments have shown that state-of-the-art CNNs failed drastically to recognize atypical objects. Table 5.1 shows the results of this experiment, where we took off-the-shelf CNNs and applied them on atypical images. The significant performance drop, when tested on atypical images, is rooted in the limited generalization power of CNN models versus the human visual system.

One might argue that this issue of cross-dataset generalization is implicitly rooted in dataset biases, and not limited to CNN models [162]. However, we argue that the huge number of labeled images in the training set of these models (here ImageNet) should alleviate this drawback. By providing a wide range of variation in terms of visual appearances of objects in training images, the effect of biases fades away. We support our argument by testing same networks on a

Method	Top-1 error (%)			Top-5 error (%)		
	Train	Test-T	Test-A	Train	Test-T	Test-A
AlexNet [89]	38.1	49.5	74.96	15.32	24.01	47.07
OverFeat [148]	35.1	45.36	75.62	14.2	22.27	46.73
Caffe [80]	39.4	51.88	77.12	16.6	24.74	46.86
VGG-16 [152]	30.9	44.04	77.82	15.3	26.31	47.49
VGG-19 [152]	30.5	43.72	76.35	15.2	26.85	45.99

Table 5.1: State-of-the-art Convolutional Neural Networks (trained on normal images) fail to generalize to atypical/abnormal images for the task of object classification. Columns "Train" show the reported errors on typical/normal images (ILSVRC 2012 validation data), while numbers in the next two columns are the errors on our atypical "Test-A", and typical "Test-T" images. The significant drops in performance, especially when tested on atypical images, show the limited generalization capacity of CNNs. Our goal is to enhance these visual classifiers and reducing this gap, without even seeing these images during the training phase.

new set of images that are disjoint from the training set of ImageNet [39], but look typical. Results of this experiment as it is reported in columns "Test-T" in Table 5.1 show a much smaller drop in accuracy, compared to the case of testing on atypical images (Test-A). We conclude that dataset bias can affect the performance of CNNs for object categorization, but it is not the main reason behind its poor generalization to new datasets.

Instead, inspired by the way humans learn object categories, we can empower CNN models with the ability to categorize extremely difficult cases of atypical images. Humans begin to form categories and abstractions at an early age[108]. The mechanisms underlying human category formation are the subject of many competing accounts, including those based on prototypes[106], exemplars[115], density estimation[6], and Bayesian inference[68]. But all modern models agree that human category representations involve subjective variations in the typicality or probability of objects within categories. In other words, typicality is a graded concept and there is no simple decision boundary between typical vs. atypical examples. A category like bird, would include both highly typical examples such as robins, as well as extremely atypical examples like penguins and ostriches, which while belonging to the category seem like subjectively "atypical" examples. Visual images can also seem atypical, in that they exhibit features that depart in some way from what is typical for the categories to which they

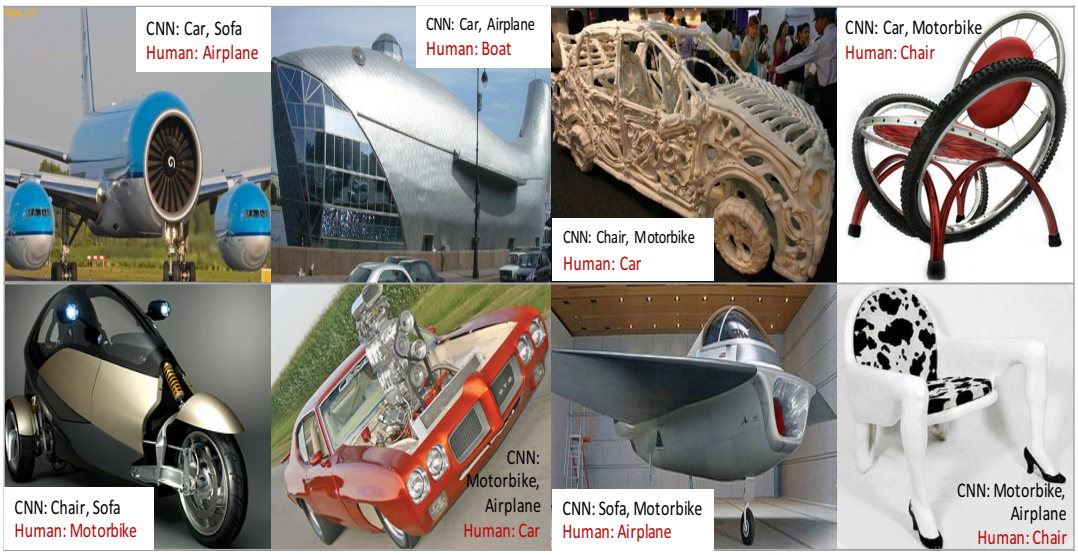

Figure 5.1: Some atypical images from "Abnormal Object Dataset" that are misclassified by a CNN object classifier (AlexNet), where as humans can categorize them correctly. Top two model predictions (in black) are reported, where the first one has 100 % model confidence.

belong. Humans learn object categories and form their visual biases by looking at typical samples [154, 130]. But they are able to generalize these visual concepts to a great extent, and recognize atypical/abnormal objects, which show significant visual variations from the training set. They achieve this ability without even observing abnormal images at the learning stage.

From computer vision and machine learning perspectives, state-of-the-art object classification and detection is based on discriminative models (e.g. SVM, CNN, Boosting) rather than generative ones. Discriminative training focuses more on learning boundaries between object classes, instead of finding common characteristics in each class. Training CNN models is based on minimization of a loss function, defined as the misclassification of training samples. In that sense, CNN implicitly emphasizes on the boundary examples rather than more representative (typical) training examples.

In this chapter, we hypothesize that not all images are equally important for the purpose of training visual classifiers, and in particular deep convolutional neural networks. Instead, we show that if training images are weighted based on how typical they look, we can learn visual classifiers with a better generalization capacity. Our final CNN model is fine-tuned only with typical images, but outperforms the baseline model (training samples are not weighted)

on dataset of atypical images. We also empirically compare a large set of functions that can be used for weighting samples, and conclude that an even-degree polynomial function of typicality ratings is the best strategy to weight training images. We also investigate the effect of loss functions and depth of network by conducting experiments on two datasets of ImageNet and PASCAL.

The main contributions of this chapter are as following:

- Evaluating CNN models on datasets of images that are different from training data, and characterizing failure cases as the poor generalization capacity of CNN models. Especially contrasting these failures to the superior performance of humans in categorizing atypical objects.

- Inspired by theories in psychology and machine learning, we propose three hypotheses to improve the generalization capacity of CNN models. These hypotheses are based on weighting training images depending on how typical they look. Our final strategy uses generative hints from prototype theory (typicality scores) to improve the generalization capacity of discriminatively trained CNN classifiers.

- We conduct an extensive set of experiments, to empirically compare different functions of typicality rating for weighting training images.

5.2 Related Work

Space does not allow an encyclopedic review of the prior literature on deep learning, but we refer interested readers to the literature review of [95]. For our research, we focus on convolutional neural networks [63, 89, 97] as the state-of-the-art deep learning models for the task of object recognition. CNN [96] has its roots in Neocognitron [62], which is a hierarchical model based on the classic notion of simple and complex cells in visual neuroscience [77]. However, CNN has additional hidden layers to model more complex non-linearities in visual data and its overall architecture is reminiscent of the LGN $\mapsto$ V1 $\mapsto$ V2 $\mapsto$ V4 $\mapsto$ IT hierarchy in the visual cortex ventral pathway. Additionally it uses an end-to-end supervised learning algorithm, called “Backpropagation” to learn weights of layers. Different variations of CNN models have

made breakthrough performance improvements in a variety of tasks in the field of computer vision.

Despite an extensive amount of prior works on applications of CNN and proposed variations of it, theoretical understanding of them remains limited. More importantly, even when CNN models achieve human-level performance on visual recognition tasks [74], what will be the difference between computer and human vision? On the one hand, Szegedy *et al.* [157] demonstrated that CNN classification can be severely altered by very small changes to images, where it leads to radically different CNN classification of images that are indistinguishable to the human visual system. On the other hand, Nguyen *et al.*[113] generated images that are completely unrecognizable by humans, but which a CNN model would classify them with 99.99% confidence. This strategy to fool CNN models, raises questions about the true generalization capabilities of such models, which we investigate it in this chapter.

In addition, recent studies in the field of neuroscience and cognition have shown the connection between deep neural networks (mainly CNN) and the visual system in human brain. Yamins *et al.*[175] showed there is a correlation (similarity) between the activation of middle layers of CNN and the brain responses in both V4 and inferior temporal (IT), the top two layers of the ventral visual hierarchy. Cadieu *et al.*[27] proposed a kernel analysis approach to show that deep neural networks rival the representational performance of IT cortex on visual recognition tasks. Khaligh-Razavi and Kriegeskorte [86] studied 37 computational model representations and found out the CNN model of [89] came the closest to explaining the brain representation. Interestingly, the amount of correlation between human IT and layers of CNN increases by moving to higher layers (fully-connected layers). They concluded that weighted combination of features of the last fully connected layer can explain IT to a full extent. It has been shown that CNN models predict human brain activity accurately in early and intermediate stages of the visual pathway [1].

There are some prior works on finding the right features [18], choosing the appropriate train set and how to order training examples for learning better classifiers [12]. Also, It has been shown that CNN models benefit from training with larger datasets of images. This is because the greatest gain in detection performance will continue to derive from improved representations and learning algorithms that can make efficient use of larger training sets [186].

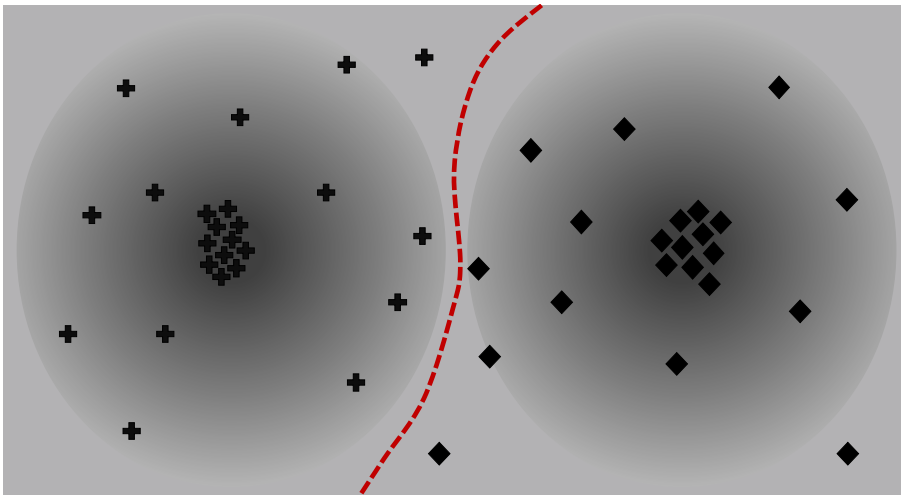

Figure 5.2: An illustration of the notion of atypical and boundary samples. Examples of two classes of cross and diamond show different shades (degrees) of typicality. While we can find the red classifier to discriminate classes, we cannot find a decision boundary between atypical vs. typical samples of the category of interest. Also, the set of samples of each class that fall close to the decision boundary (boundary examples) does not include all atypical examples.

However, this leaves open the question if training images should be equally weighted during the training or not?

5.3 Computational Framework

In this section, we first review some through theoretical background and compelling theories about the learning of visual concepts in both fields of psychology and computer vision. We explain the role of atypical examples in training classifiers, and how one can measure the typicality of objects in an image. Then we propose three hypotheses to use these typicality scores for improving the generalization capacity of visual classifiers.

5.3.1 Framework Motivation

Humans learn a visual object class by looking at examples that are more representative for that object category, or what is called typical samples [154, 130]. It has been shown that children who learn a category by looking at more typical samples, later can recognize its members better [132]. If training examples look more typical, they fall close to each other in an underlying

space of visual features. This learning strategy not only helps humans to form a concept, but also allows them to more easily apply the learned concept to novel images. This great ability of human visual system allows them to recognize completely different variations of an object, even to the extent of atypical ones. This suggests that emphasizing on typical examples might be helpful for improving the generalization capacity of classifiers.

However, state-of-the-art object classifiers in computer vision are discriminative models, where they distinguish different objects by learning category boundaries. CNN models as discriminative deep neural networks have multiple layers to learn a hierarchy of visual features and categorize objects by minimizing a loss function, which is based on misclassification errors. In other words, if an image is classified correctly (usually the case for typical images), it has little or no impact on the loss function, hence can be ignored in the training phase. This implies that examples close to the decision boundary, which are likely to be more atypical images, play a substantial role in learning CNN models. This suggests that CNN training emphasizes on more atypical images to learn visual classifiers with a better performance.

We illustrates the connection between typical, atypical, boundary and misclassified training samples in Figure 5.2; where examples of two object classes ($\mathcal{C}$) are shown with diamonds and crosses, and the red dotted line is one possible decision boundary. There are two main points to be taken from this illustration:

First, as we discussed in Section 5.1 typicality is a graded concept, which directly relates to the likelihood of an observation given its class distribution $\mathcal{P}(\mathcal{X}|\mathcal{C})$. Very typical examples are expected to be located close to the mean of each class distribution (center of clouds), with a high probability [57]. Moreover, as we move away form the center, we still observe examples of the same category. But every member of the category shows a different rate of typicality $\mathcal{P}(\mathcal{X}|\mathcal{C})$. This is visualized as a smooth transition when moving away from the center of a class. More importantly there is no clear boundary between typical and atypical members.

Second, atypicality happens for a variety of reasons. This is visualized as there is not a unique axis for transition from darker to brighter shades of gray. Although examples close to the decision boundary might be atypical for their category; but the atypical examples are more diverse and not limited to the boundary examples. In conclusion, the two sets of atypical and boundary examples are not equal.

5.3.2 Sample-Based Weighted Loss

CNN architecture consists of multiple blocks, where each block has a convolution layer, possibly followed by pooling and normalization layers. On the top of these blocks, there are fully connected layers that are designed to learn more complex structures of object categories. The last layer of CNN computes the "loss" as a function of mismatch between the model prediction and the ground truth label. The training of CNN is formulated as minimization of this loss function [96]. However, our work is the first study to analyze the effect of weighting samples and using different loss functions incorporating in typicality scores, to improve generalization capacity of CNN. We associate each sample $\mathcal{X}$ with a weight τ as a function of its typicality, which we explain later. We build our models based on two loss functions: *Softmax log* and *Multi-class structured hinge*. While the first one is the fastest and widely used in prior works, the later takes into account all the possible category memberships for a given object.

Softmax log loss:

For classification problems using deep learning techniques, it is common to use the softmax of one of the $\mathcal{C}$ encodings at the top layer of the network, where $\mathcal{C}$ is the number of classes. Assuming the output to the *i-th* node in the last layer, for the image $\mathcal{X}$ is: $z_i(\mathcal{X})$. Then our goal is to minimize the weighted multinomial logistic loss ($\mathcal{L}$) of its softmax over N training images :

$$\mathcal{L} = \sum_n -\tau(\mathcal{X}_n) * \log(\sigma_i(\mathcal{X}_n)) \quad (n = 1, ..., N)$$

$$\sigma_i(\mathcal{X}_n) = \exp(z_i(\mathcal{X}_n))/\sum_j \exp(z_j(\mathcal{X}_n)), (i, j = 1, ..., \mathcal{C}).$$

Multi-class structured hinge loss:

It is also known as the Crammer-Singh loss, and is widely used for the problem of structured prediction. This loss function is similar to hinge-loss, but it is computed based on the margin between the score of the desired category and all other prediction scores ($\phi(i)$) [34]. We

Loss	Test set	Typ	Atyp	Cls-Typ	Cls-Atyp
MS-Hinge	Atypical	68.58	70.64	70.84	68.47
Softmax	Atypical	63.69	66.82	65.81	66.48
MS-Hinge	Typical	79.90	84.07	82.88	83.40
Softmax	Typical	77.11	80.42	83.40	82.96

Table 5.2: Object classification accuracy (%) of the AlexNet on two test sets of Typical(lower box) and Atypical(upper box) images. Two loss functions (rows) are compared, when training samples are weighted via four functions (columns): Raw score of Typicality (first), Raw score of Atypicality (second), Class-specific typicality (third) and Class-specific atypicality (fourth).

aggregate this loss function ($\mathcal{L}$) by a weighted summation over training samples:

$$\mathcal{L} = \sum_{n} \tau(\mathcal{X}_n) * \max(0, 1 - \phi_i(\mathcal{X}_n))$$

$$\phi_i(\mathcal{X}_n) = z_i(\mathcal{X}_n) - max_{i \neq j}(z_j(\mathcal{X}_n)).$$

Multi-class hinge loss is particularly of our interest as it considers the margin between all class predictions. This is an important piece of information when we want to generalize the learned visual classifiers to the case of atypical objects. These examples are harder to categorize, and class prediction is not a distribution with its peak around the desired class. In fact, the object might get high class confidence for multiple categories, which results in a smaller ϕ and bigger $\mathcal{L}$.

5.3.3 Measuring Typicality of Objects

We have two approaches for measuring the typicality of objects. On the one hand, we compute the probability score $\mathcal{P}(T|\mathcal{X})$ as how typical (T) is the object only based on its visual features $\mathcal{X}$. For the case of class-specific typicality we can infer: $\mathcal{P}(T|\mathcal{X}) \propto \mathcal{P}(X|\mathcal{C})$ where $\mathcal{C}$ indicates the category, and independent of the class: $\mathcal{P}(T|\mathcal{X}) \propto \mathcal{P}(\mathcal{X})$. Then its complement $(1 - \mathcal{P}(T|\mathcal{X}))$ is the probability of atypicality.

To implement this probability, we use one-class SVM where only positive samples of one category (here typical images) are used and there is no negative (atypical) training example. This model can be understood as a density estimation model where there is no prior knowledge about the family of the underlying distribution. We learn this one-class SVM in two scenarios: 1) General class-independent typicality: all images are used; 2) Class-specific typicality: for

each category one SVM is trained only based on typical images of the category of interest. We refer to these models as "external score of typicality". This is because these scores are computed using a model distinct from object classifier (here CNN), and based on visual features different from what we use for object categorization. These scores are computed offline for all training images and not changing over different epochs of CNN training.

On the other hand, we can judge typicality of training images directly from the output of CNN visual classifiers. Lake *et al.*[93] showed that the output of the last layer of CNN models can be used as a signal for how typical an input image looks like. In other words, typicality ratings are proportional to the strength of the classification response to the category of interest. Assuming the classification loss is defined over C object categories and there are N nodes in the last layer, we compute "internal probability of typicality" as:

$$Z_i = \exp(y_i)/\sum_{j=1}^{C}\exp(y_j); \quad where: y_j = \sum_{i=1}^{N} x_i W_{ij} \tag{5.1}$$

Alternatively, we use the entropy of a category prediction as a measure of uncertainty in responses, which punishes more uncertain classifications. We call this "internal entropy of typicality" and compute it as : $-Z_i log(Z_i)$.

5.3.4 Hypotheses

We propose three hypotheses to improve the generalization of visual classifiers, especially when the test image looks substantially different(atypical) from training images:

First, Inspired by the prototype theories from psychology, we hypothesize that learning with more emphasis towards representative (typical) samples would increase the generalization capacity of the visual classifier.

Second, Learning with emphasis on more atypical examples in the training set would enhance the generalization capacity. This is because it complements the way that loss function emphasizes boundary examples. This hypothesis, places additional emphasis on other possible directions of atypicality in training data that might not be on the boundary.

Third, We hypothesize that emphasizing on both typical and atypical examples might be the key for a better generalization performance, and should be used for learning visual classifiers. The main idea behind this hypothesis is the fact that any visual classifier should

learn how the object category is formed (mainly typical examples), and how much a variation it would allow for its members (atypical samples).

To implement the first two hypotheses we multiply the loss of each sample by $\tau(\mathcal{X})$, which is a function of typicality (for the first hypothesis) or atypicality (second hypothesis). To investigate the effect of different functions of the typicality score, we evaluate exponential ($\exp \mathcal{P}(T|\mathcal{X})$) and gamma ($\gamma^{\mathcal{P}(T|\mathcal{X})}$) functions to emphasize typicality versus a logarithmic function ($-\log(\mathcal{P}(T|\mathcal{X}))$) to emphasize atypicality. This helps us to evaluate the generalization capacity of a CNN model, when trained with non-linear weighting. We evaluate our last hypothesis by implementing the weighting function as an even-degree polynomial:

$$\mathcal{F}(T) = \alpha(T - \mu)^d + \beta; \quad d = 2k(k = 1, ..., n) \tag{5.2}$$

These functions are symmetric around the average typicality score in the dataset (μ), and place more emphasis on data points in both extremes of the typicality axis.

5.4 Experimental Results

Datasets:

We used three image datasets: 1) ImageNet challenge (ILSVRC 2012 & 2015), 2) Abnormal Object Dataset [146], 3) PASCAL VOC 2011 train and validation set. We conducted our experiments with six object categories: Aeroplane, Boat, Car, Chair, Motorbike and Sofa. We did this to be able to verify our generalization enhancement for atypical images in Abnormal Objects dataset, which contains these categories. We merged related synsets of ILSVRC 2012 to collect 16153 images of these categories, which we refer to as "train set I".

Additionally, we experimented with train and validation set of PASCAL 2011. This is needed because due to a higher level of supervision in PASCAL data collection process, images are more likely to look typical. However, ImageNet data shows significant variations in terms of visual appearance (pose, missing or occluded parts, etc.) that can make the image and object look less typical. We collected 4950 images from PASCAL dataset, which we refer to as "train set II".

We also used a subset of 8570 images from ILSVRC 2015 detection challenge, which we

Weighting Function used in Fine-Tuning	Mean Accuracy (%)			
	Test Atypical		Test Typical	
	Epoch 1	Epoch 10	Epoch 1	Epoch 10
No weight	56.39	65.18	78.15	83.51
Random	57.15	66.45	73.60	83.84
Typicality	64.53	68.58	69.22	79.90
Atypicality	66.61	70.65	75.82	84.07
Cls-Typ	67.25	70.84	77	81.88
Cls-Atyp	63.26	68.46	76.96	83.40
Log-Typ	64.38	68.28	78.80	83.67
Log Cls-Atyp	64.21	67.80	76.13	83.24
Memorability	64.69	68.33	76.31	83.96
Poly Deg-2	59.13	69.49	80.03	84.42
Poly Deg-4	60.22	71.52	77.74	83.45
Poly Deg-6	60.86	70.31	77.66	84.22
In-Probability	65.97	69.53	80.71	85.82
In-Entropy	60.54	68.05	79.44	82.29
In-Prob + Atyp	62.94	68.21	75.82	83.09

Table 5.3: Object classification performance with AlexNet fine-tuned on "Train Set I". MS-Hinge loss is used and rows show different sample-based weighting functions of typicality/atypicality. Average variance of response of these accuracies is 0.03

call "test typical", and are completely disjoint from the set used in training ("train set I). Images of [146] form our "test atypical" set, which contain confirmed atypical/abnormal objects.

Typicality estimation:

We measured the typicality of images via one-class SVMs in two settings: General and Class-specific. The first case is independent of the object-category and only measures how typical the input image looks in general. But, for the latter we trained six (one for each category) one-class SVMs with typical images of the category of interest. We extracted kernel descriptors of [19] at three scales as the input features.

Visual classifier:

We investigated our three hypotheses using the CNN model of AlexNet [89]. Nevertheless, our approach can be incorporated in other state-of-the-art CNN models for object classification as well. We acquired the Caffe implementation [80] and fine-tuned the network for all the

following experiments. For the final fine-tuning of the model, although the training strategy is still discriminative, but typicality of the training samples will influence the major parameter estimation.

5.4.1 Comparison of Loss Functions

To find the proper loss function for fine-tuning the network, we conducted an experiment with two losses: Softmax and Multi-structured hinge (MS-Hinge). For this experiment we only fine-tuned the last fully-connected layer with "Train Set I". Table 5.2 shows the performance comparison based on using different loss functions and sample-based weighting methods. We conclude that independent of the weighting strategy, Multi-structured hinge (MS-Hinge) performs better than the Softmax loss. Consequently, the rest of experiments were conducted based on fine-tuning with MS-hinge loss.

5.4.2 Comparison of Weighting Functions

We conducted a set of experiments to compare the performance of CNN models for the task of object classification, when fine-tuned using different weighting functions. Table 5.3 shows the result of these experiments on the two test sets of Typical and Atypical. We report the mean accuracy after the first and tenth epochs. While the result of the first epoch indicates how fast the network can learn a category, the tenth epoch elaborates the performance when the network has matured (trained for a longer time).

External score of typicality:

The first box in Table 5.3 shows the baseline experiments, when the first row is fine-tuning the AlexNet without any sample-based weighting. Second row shows weighting training images with a random number between zero and one. Comparing this row with the case of not weighting samples, shows there is almost no increase in the performance, and even decreasing when tested on typical images. This verifies that randomly weighting training data does not help improving the generalization capacity of the trained network.

Weighting Function used in Fine-Tuning	Mean Accuracy (%)			
	Test Atypical		Test Typical	
	Epoch 1	Epoch 10	Epoch 1	Epoch 10
No weight	30.03	48.40	51.22	64.17
Random	29.22	49.18	49.03	58.9
Memorability	35.94	47.12	54.28	69.15
Typicality	29.71	47.76	48.12	61.55
Atypicality	41.21	52.24	55.3	70.28
Log-Atyp	37.38	45.69	51.37	62.46
Log-Typ	36.95	50.80	52.76	68.88
Poly Deg-2	41.37	55.44	54.8	73.02
Poly Deg-4	42.33	56.39	53.9	72.42
Poly Deg-6	44.73	52.72	52.93	72.7

Table 5.4: Object classification performance with AlexNet fine-tuned on "Train Set II" (PASCAL dataset). MS-Hinge loss is used and rows show different sample-based weighting functions of typicality/atypicality. Average variance of response of these accuracies is 0.07

Next box represents the results of using the typicality or atypicality score (the output probability of one-class SVM) for weighting training images. We conclude that fine-tuning with raw atypicality/typicality weighting can significantly enhance the generalization of CNN, even after the first epoch. However, fine-tuning with raw typicality can degrade the performance, when tested on typical images. The third box has similar results, where typicality or atypicality are computed for each object-class separately, based on the class-specific one-class SVMs.

Fourth box in Table 5.3 investigates the importance of non-linear weighting functions. First and second row are the results of using logarithmic functions, where $\tau()$ is either typicality score (first row) or class-specific atypicality scores (second row). We conclude that networks do not gain much from non-linear functions of either typicality or atypicality scores, when test on atypical images. But non-linearities help stabilizing the performance on typical images. The last row of the fourth box, indicates that fine-tuning AlexNet with the memorability score [87] will increase its generalization performance (comparing to baselines). However, fine-tuning with memorability do not outperform typicality weightings.

The fifth box in Table 5.3 evaluates our third hypothesis, where three polynomials are used for weighting the training samples. In general, this strategy outperforms other methods (comparing the tenth epoch performance) on atypical test set, and comparing to the baseline improves the performance on the typical set as well.

Layers changed in fine-tuning	Image Set Used in Test	Weighting Functions Used in Fine-Tuning			
		Atyp	Typ	Log-T	Ploy2
Top 2 FC	Atypical	68.17	64.69	67.41	69.97
Top 3 FC	Atypical	66.13	51.28	68.37	69.33
Top 2 FC	Typical	81.19	79.52	80.6	82
Top 3FC	Typical	78.51	77.1	76.13	79.41

Table 5.5: Evaluation of the effect of depth for generalization of AlexNet. Comparison of two alternative models, when we go deeper than the first fully connected layer. One with changing top two and the other one with fine-tuning top three fully connected layers. Models are fine-tuned with "Train Set I" and MS-Hinge loss is used.

Internal score of typicality:

The last box in Table 5.3 have the classification performance when networks are fine-tuned with an internal signal of typicality. These scores can be either normalized class predictions, or what we call "internal probability of typicality" as it is in the first row; Or "internal entropy of class distribution" in the second row. The last experiment (row) follows a hybrid approach, which in the first epoch samples are weighted with atypicality scores (from one-class SVM), and starting the second epoch, samples are weighted with internal scores.

Experiment with fine-tuning on PASCAL:

We recompiled previous experiments when networks were fine-tuned on "Train Set II" (PASCAL images). These results (Table 5.4) verify our hypothesis that we can enhance the generalization capacity of CNN with weighting training examples based on functions of the typicality scores. Interestingly, we gained bigger performance improvements (from the first epoch to the tenth epoch) when fine-tuned on PASCAL, rather than ImageNet . We relate this to the more diverse visual appearance and higher noise in ImageNet collection.

5.4.3 Investigation of The Effect of Depth

We investigated the importance of fine-tuning deeper layers of CNN, to train models with a better generalization capacity. Table 5.5 shows the results of fine-tuning top-two or top-three fully connected layers of AlexNet. In the first row of each box, we changed the FC7 to have

2048 nodes. Similarly in the second row of each box, we halved the number of nodes in both FC6 and FC7. In all three models (including one reported in previous sections), we used MS-hinge loss to learn the parameters of the network. These experiments show that going deeper would hurt the fine-tuned network when tested on atypical images. We would partially relate this to the limited number of images that are available for fine-tuning, therefore the network overfits to the training date (ImageNet). Digging deeper into this experiment with more training examples is considered as the future work. Also we believe changing the loss function at the time of fine-tuning (as it is in our case) would not be beneficial when we consider deeper layers.

5.5 Conclusion

In this chapter, we conducted a study on the generalization capacity of convolution neural networks. There are several points that we can conclude from this study. The state-of-the-art CNN object classifiers fail drastically when they are applied on atypical images. Atypicality is not necessarily equivalent to samples on the boundary, which common loss functions try to emphasize in learning. However, atypical images show extreme changes in visual features, which are still understandable to the human visual system.

The main result of this chapter is that involving information about the typicality/atypicality of training samples as a weighting term in the loss function helps greatly in enhancing the performance on unseen atypical examples, when training only using typical examples. We proposed different ways to achieve this weighting of samples based on external (from the sample distribution) and internal signals to the network. We also found that symmetrically weighting highly typical and highly atypical examples in training gives better generalization performance. We believe that this is because the typicality/atypicality scoring of the data include information about the distribution of the samples, and therefore it incorporates in generative "hints" to the discriminative classifier.

The typicality weighting not only helps the generalization, but also helps faster learning where the network was shown to converge to significantly better results after a single epoch. For the future work, we plan to design new loss functions that can benefit more from measuring typicality of images. Also, investigation of applicability of this framework (using typicality

weighting in training) for the case of image captioning is considered as another interesting future work.

Chapter 6

Expanded Visual Knowledge

6.1 Fine-Grained Object Categorization via Localized Attribute Detection

In this work we propose an end-to-end system for detection and classification of fine-grained objects. Our approach takes the advantage of part-based models to capture the variation in the object structure and use attributes for describing objects in fine details. We acquire strongly supervised part-based models to detect objects as well as their parts. We augment object parts in our model with a set of localized attributes corresponding to shape, color and texture. We use this part-specific attribute-based representation to classify fine-grained object categories. Our experiments on fine-grained object classes in Caltech UCSD Birds data set [169], show that our attribute-based representation outperform prior work [183], which only uses low-level visual features to classify the object.

6.1.1 Introduction

Fine-grained categorization refers to the problem of subordinate classification of objects, where the hierarchy of object classes is either natural (e.g.breeds of dogs, species of birds) or artificial (e.g. different types of airplanes). Despite the basic-level categories (e.g. car, cow), fine-grained objects (e.g. different types of cars) have highly similar body configuration, which makes it hard to distinguish them solely based on the shape information. For instance, a cow and a car can be easily distinguished by representing each one with an ensemble of parts, which clearly differs from one to another. But different categories of cars (e.g. sedan, SUV or sport) share the same set of parts and body pose [24, 40]. This clarifies that appearance information (texture, color, etc.) plays a substantial role for categorization of fine-grained object classes [182]. Additionally, these visual attributes can be localized and assigned to

different parts of the object [42].

In this work, we propose a model for categorization of fine-grained objects based on part-based visual features and attributes. On one hand, our model detects objects and predicts their parts. On the other hand, it classifies the detected object based on its localized attributes and visual appearance. In order to detect objects, we train strongly supervised deformable part models (DPM) for fine-grained object categories. These models predict the location of the object along with its parts. Having these predicted parts, we find a set of localized attributes that are unique for each part. We categorize the detected object based on the part-based visual features and localized attributes. In our experiments, we show the value of adding localized attributes to improve the performance of classifiers.

6.1.2 Proposed Model

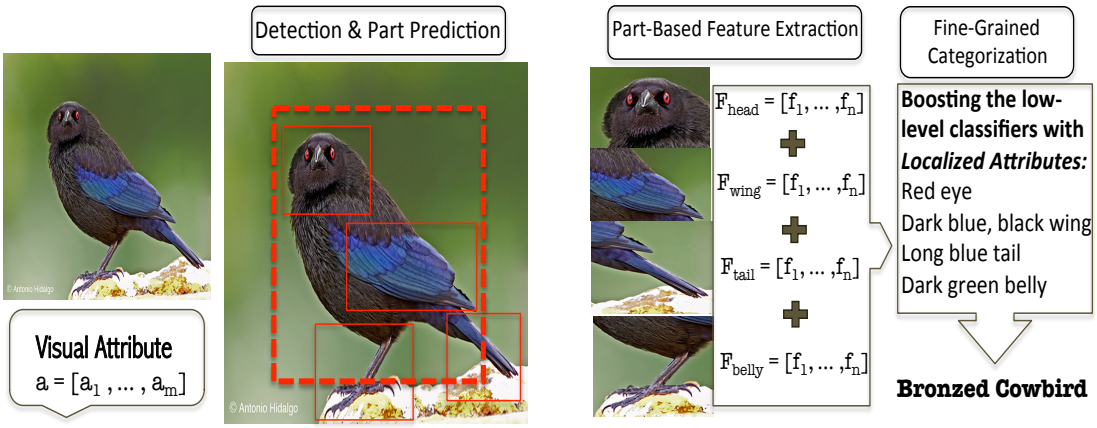

Figure 6.1: Illustration of our framework for "Detection" and "Classification" of "Fine-Grained object categories". Given an image, our model first detects the object and predicts its parts. Next, it classifies the detected object based on concatenation of visual features and a set of localized attributes that are extracted for each part separately.

"Deformable Part Model" ("DPM" [58]) is a powerful model for object detection, which is capable of handling different body configurations of a given object category. DPM trains an object detector, which has a root filter and a set of discriminately discovered parts that are connected to the root following some geometric constraints. However, these predicted parts do not necessarily correspond to meaningful parts of the object(e.g. wing of a bird). Azizpour and

Laptev [7] proposed a strongly supervised variation of DPM ("SSDPM"), which overcomes this issue by taking advantage of part annotations during training. In this work, we use SSDPM to train detectors for fine-grained object categories, which is more challenging compared to the original setting of [7].

As depicted in figure 6.1, for a given image, we detect the object and predict its four parts("Head", "Wing", "Tail" and "Belly"). In the next step, we extract a compact set of low-level visual features from each part of the object separately and concatenate them to get one feature vector for the whole object. We also augment these part-based visual descriptors with a set of "Localized Attributes" that are extracted for each part. In contrast to previous work on localized attributes, our model does not require human supervision for learning attributes of each object. We use category-level text description to learn attributes for each part. Finally, we use these visual features and attributes to classify the detected object based on the confidences of a set of one vs. all classifiers.

It is been shown that detection of "Localized Attributes" plays an important role for discriminating fine-grained categories [42]. The importance of location for localized attributes is crucial for fine-rained categories. For example, we can describe a crow as a black bird. But for distinguishing different species of birds from crows, we should emphasize on the location that black is the dominant attribute (e.g. head vs. wing). Duan et al. [42] use human intervention for training localized attributes, which might be expensive or unavailable. This justifies the importance of our system that can automatically determine localized attributes. Additionally, attribute-based object classification gives us the opportunity of zero-shot learning of a new category. Although majority of previous work on zero-shot learning use globally defined attributes (e.g. red or stripped), our model uses localized attributes (e.g. stripped wing).

6.1.3 Experiments

We used both versions of Caltech-UCSD Birds data set [169] to conduct our experiments. CUB200-2011 has 11788 images of 200 bird species in North America. All images are annotated with object bounding box, keypoint locations and 312 visual attributes. In order to find ground truth bounding box for each part of birds, we define an one-to-one mapping between keypoints and parts. For example, "beak" and "eye" are two keypoints, which are exclusively

assigned to the part "Head". Having this mapping we locate a part by fitting a reasonable size bounding box around its corresponding keypoints. As each keypoint is assigned to one and only one part, there is not any overlap between these ground truth bounding boxes. We consider a part missing, if half of its keypoints are invisible according to the visibility term in the annotation.

Table 6.1: Localized Attribute Prediction Accuracy

Part	Mean Accuracy (%)
Wing	87.41
Head	87.59
Tail	86.36
Belly	87.46
Full body	87.59

For the task of object detection, we trained strongly supervised deformable part models [7] for categories of birds in CUB200-2011 using four predefined parts:"Wing", "Head","Tail" and "Belly". Training a model with so many components or so many models with few components per a model (e.g. one model for each of 200 subordinate categories of Birds) is impractical as number of samples for each component will be limited. As a result, we trained 20 individual models with two components as following: Initially we clustered training images based on their pose, viewpoint and bounding box ratio and trained one model for each cluster. These models implicitly capture some hierarchy for Birds. This can be seen by looking at the output of the clustering step and noticing the fact that usually all of the subspecies of one category of birds fall in one cluster. Then we train an individual model for each of these clusters with few components. These components will capture variations of shape and pose for a subset of birds. The described model has the average precision of 31.4% for the task of detection.

We extracted kernel descriptors [19] for each detected part using Gradient, Local Binary pattern, RGB color and Normalized RGB color kernels. We quantized the extracted feature vectors at one scale to get a 4000 dimensional feature vector for each part and whole object, resulting a $20,000$ dimensional feature vector for each image. We also extracted a set of localized attributes, exclusively defined for each part. Number of attributes per part, ranges from 34 attributes (for the whole body) to maximum of 112 attributes (for "Head"). We train a linear SVM for each attribute using positives and negative samples taken from the same part across all

Table 6.2: Categorization results on CUB200-2011

Method	Mean Accuracy (%)
PPK [182]	28.18
PPK with KDES [182]	28.20
Kernel Descriptors [19]	42.53
Template Matching [178]	43.67
DPD-strong-2 [183]	50.05
DPD-Weak-8 [183]	50.98
Ours-Localized Attributes	33.8
Ours without Localized Attributes	52.2
Ours with Localized Attributes	**56.45**
Oracle	64.53

the categories. The importance of each dimension of the feature vector is determined by using $l1$-regularization. Taking union over all the important dimensions given by $l1$-regularization step followed by training attribute classifiers over these dimensions results in more robust attribute classifier. Table 6.1 shows the average accuracy of these part-based attribute classifiers for images in CUB200-2011. Each row corresponds to the average accuracy of attribute classifiers for that specific part, where the goal is to predict the presence of attributes in test images. Also for the task of fine-grained categorization we trained a linear SVM on top of attribute vectors. This experiment is called "Ours-Localized Attributes" in the 7-th row of table 6.2. Although attribute vectors are significantly smaller than the raw low-level feature vectors (312 vs. $20,000$); but the accuracy of the model trained on attributes is lower than one trained on original features (next row in the same table). The final model (last row) uses both kernel descriptors and localized attributes to outperform previous methods. This shows the advantage of using localized attributes for boosting classifiers that are trained on low-level visual features.

The work of [183] is the closest to ours, but our model is different as following: 1) It is fully supervised across all the experiments rather weakly supervised (one used in [183]). 2) Our model is trained with four detailed parts instead of two parts, which convey more semantic of part based models. As extracted features are part-based, having more accurate parts result in a better feature extraction. The fact that quality of part-based feature extraction is directly related to the quality of detected parts can demonstrated by comparing our results without localized attributes (8-th row in table 6.2) and one reported by Zhang et al. [183] in rows $5\&6$).

The data set of CUB200-2010 has fewer images (6033 in total) from the same categories of birds. This version comes with object bounding box only and does not have part annotation. Also, the provided list of visual attributes in this version is smaller (288), and not all of them are localized. As the needed annotation (part location) is only provided with the 2011 version, we used the train set of this version to train our model for detection. Because attribute annotations are provided for each image, we aggregate scores for all images in a category to get one single attribute vector as a descriptor for the category. These attribute vectors have continuous values that can be turned into binary by setting a threshold. Table 6.3 shows the result of object categorization for images in this data set.

Table 6.3: Categorization results on CUB200-2010

Method	Mean Accuracy (%)
MKL [24]	19.0
Random Forest [180]	19.2
Kernel Descriptors [19]	26.4
TriCos [30]	26.7
Template Matching [178]	28.2
Segmentation [3]	30.2
Bubblebank [40]	32.5
DPD-strong-2 [183]	34.5
Ours without Localized Attributes	33.8
Ours with Localized Attributes	**37.1**

6.2 Zero-shot learning of object categories via text description

The main question we address in this paper is how to use purely textual description of categories with no training images to learn visual classifiers for these categories. We propose an approach for zero-shot learning of object categories where the description of unseen categories comes in the form of typical text such as an encyclopedia entry, without the need to explicitly defined attributes. We propose and investigate two baseline formulations, based on regression and domain adaptation. Then, we propose a new constrained optimization formulation that combines a regression function and a knowledge transfer function with additional constraints to predict the classifier parameters for new classes. We applied the proposed approach on two fine-grained categorization datasets, and the results indicate successful classifier prediction.

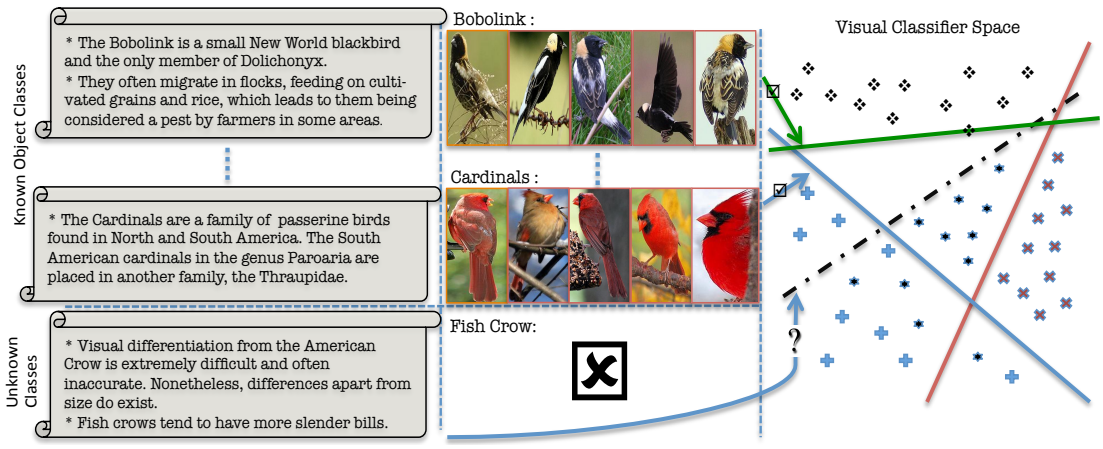

Figure 6.2: Problem Definition: Zero-shot learning with textual description. Left: synopsis of textual descriptions for bird classes. Middle: images for "seen classes". Right: classifier hyperplanes in the feature space. The goal is to estimate a new classifier parameter given only a textual description

6.2.1 Introduction

One of the main challenges for scaling up object recognition systems is the lack of annotated images for real-world categories. Typically there are few images available for training classifiers for most of these categories. This is reflected in the number of images per category available for training in most object categorization datasets, which, as pointed out in [137], shows a Zipf distribution. The problem of lack of training images becomes even more severe when we target recognition problems within a general category, *i.e.* , fine-grained categorization, for example building classifiers for different bird species or flower types (there are estimated over 10000 living bird species, similar for flowers). Researchers try to exploit shared knowledge between categories to target such scalability issue. This motivated many researchers who looked into approaches that learn visual classifiers from few examples, e.g. [38, 55, 10]. This even motivated some recent work on zero-shot learning of visual categories where there are no training images available for test categories (unseen classes), *e.g.* [94]. Such approaches exploit the similarity (visual or semantic) between seen classes and unseen ones, or describe unseen classes in terms of a learned vocabulary of semantic visual attributes.

In contrast to the lack of reasonable size training sets for a large number of real world categories, there are abundant of textual descriptions of these categories. This comes in the form of dictionary entries, encyclopedia articles, and various online resources. For example, it

is possible to find several good descriptions of a "bobolink" in encyclopedias of birds, while there are only a few images available for that bird online.

The main question we address in this chapter is how to use purely textual description of categories with no training images to learn visual classifiers for these categories. In other words, we aim at zero-shot learning of object categories where the description of unseen categories comes in the form of typical text such as an encyclopedia entry. We explicitly address the question of how to automatically decide which information to transfer between classes without the need of human intervention. In contrast to most related work, we go beyond the simple use of tags and image captions, and apply standard Natural Language Processing techniques to typical text to learn visual classifiers.

Similar to the setting of zero-shot learning, we use classes with training data (seen classes) to predict classifiers for classes with no training data (unseen classes).Recent works on zero-shot learning of object categories focused on leveraging knowledge about common attributes and shared parts [94]. Typically, attributes [146, 53] are manually defined by humans and are used to transfer knowledge between seen and unseen classes. In contrast, in our work we do not use any explicit attributes. The description of a new category is purely textual and the process is totally automatic without human annotation beyond the category labels.

The contribution of the chapter is on exploring this new problem, which to the best of our knowledge, is not explored in the computer vision community. We learn from an image corpus and a textual corpus, however not in the form of image-caption pairs, instead the only alignment between the corpora is at the level of the category. We propose and investigate two baseline formulations based on regression and domain adaptation. Then we propose a new constrained optimization formulation that combines a regression function and a knowledge transfer function with additional constraints to solve the problem.

Beyond the introduction and the related work sections, the rest of this chapter is structured as follows: Sec 6.2.3 introduces the problem definition and proposed baseline solutions. Sec 6.2.6 describes the solution framework. Sec 6.2.7 explains the experiments performed on Flower Dataset [114] (102 classes) and Caltech-UCSD dataset [173] (200 classes).

6.2.2 Related Work

Our proposed work can be seen in the context of knowledge sharing and inductive transfer. In general, knowledge transfer aims at enhancing recognition by exploiting shared knowledge between classes. Most existing research focused on knowledge sharing within the visual domain only, *e.g.* [70]; or exporting semantic knowledge at the level of category similarities and hierarchies, *e.g.* [59, 137]. We go beyond the state-of-the-art to explore cross-domain knowledge sharing and transfer. We explore how knowledge from the visual and textual domains can be used to learn across-domain correlation, which facilitates prediction of visual classifiers from textual description.

Motivated by the practical need to learn visual classifiers of rare categories, researchers have explored approaches for learning from a single image (one-shot learning [104, 55, 61, 10]) or even from no images (zero-shot learning). One way of recognizing object instances from previously unseen test categories (the zero-shot learning problem) is by leveraging knowledge about common attributes and shared parts. Typically an intermediate semantic layer is introduced to enable sharing knowledge between classes and facilitate describing knowledge about novel unseen classes, *e.g.* [118]. For instance, given adequately labeled training data, one can learn classifiers for the attributes occurring in the training object categories. These classifiers can then be used to recognize the same attributes in object instances from the novel test categories. Recognition can then proceed on the basis of these learned attributes [94, 53]. Such attribute-based "knowledge transfer" approaches use an intermediate visual attribute representation to enable describing unseen object categories. Typically attributes are manually defined by humans to describe shape, color, surface material, *e.g.* , furry, striped, *etc.* Therefore, an unseen category has to be specified in terms of the used vocabulary of attributes. Rohrbach *et al.* [131] investigated extracting useful attributes from large text corpora. In [119], an approach was introduced for interactively defining a vocabulary of attributes that are both human understandable and visually discriminative. In contrast, our work does not use any explicit attributes. The description of a new category is purely textual.

The relation between linguistic semantic representations and visual recognition have been explored. For example in [38], it was shown that there is a strong correlation between semantic

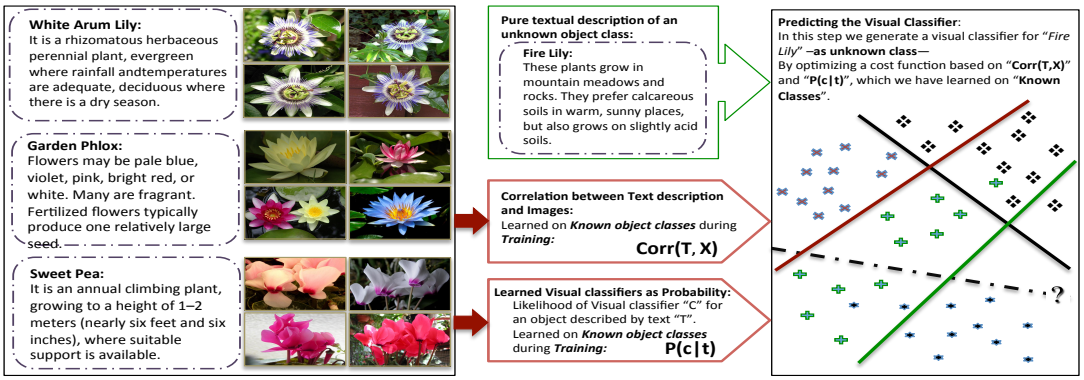

Figure 6.3: Illustration of the Proposed Solution Framework for the task Zero-shot learning from textual description.

similarity between classes, based on WordNet, and confusion between classes. Linguistic semantics in terms of nouns from WordNet [105] have been used in collecting large-scale image datasets such as ImageNet[39] and Tiny Images [163]. It was also shown that hierarchies based on WordNet are useful in learning visual classifiers, *e.g.* [137].

One of the earliest work on learning from images and text corpora is the work of Barnard *et al.* [9], which showed that learning a joint distribution of words and visual elements facilitates clustering the images in a semantic way, generating illustrative images from a caption, and generating annotations for novel images. There has been an increasing recent interest in the intersection between computer vision and natural language processing with researches that focus on generating textual description of images and videos, *e.g.* [54, 91, 179, 88]. This includes generating sentences about objects, actions, attributes, patial relation between objects, contextual information in the images, scene information, *etc.* In contrast, our work is different in two fundamental ways. In terms of the goal, we do not target generating textual description from images, instead we target predicting classifiers from text, in a zero-shot setting. In terms of the learning setting, the textual descriptions that we use is at the level of the category and do not come in the form of image-caption pairs, as in typical datasets used for text generation from images, *e.g.* [117].

6.2.3 Problem Definition

Fig 6.2 illustrates the learning setting. The information in our problem comes from two different domains: the visual domain and the textual domain, denoted by $\mathcal{V}$ and $\mathcal{T}$, respectively. Similar to traditional visual learning problems, we are given training data in the form $V = \{(x_i, l_i)\}_N$, where x_i is an image and $l_i \in \{1 \cdots N_{sc}\}$ is its class label. We denote the number of classes available at training as N_{sc}, where sc indicates "seen classes". As typically done in visual classification setting, we can learn N_{sc} binary one-vs-all classifiers, one for each of these classes. Let us consider a typical binary linear classifier in the feature space in the form $f_k(\mathbf{x}) = \mathbf{c}_k^T \cdot \mathbf{x}$ where $\mathbf{x}$ is the visual feature vector amended with 1, and $\mathbf{c}_k \in \mathbb{R}^{d_v}$ is the linear classifier parameters for class k. Given a test image, its class is determined by$l^* = \arg\max_k f_k(\mathbf{x})$ Our goal is to be able to predict a classifier for a new category based only on the learned classes and a textual description(s) of that category. In order to achieve that, the learning process has to also include textual description of the seen classes (as shown in Fig 6.2). Depending on the domain we might find a few, a couple, or as little as one textual description to each class. We denote the textual training data for class k by $\{t_i \in \mathcal{T}\}^k$. In this paper we assume we are dealing with the extreme case of having only one textual description available per class, which makes the problem even more challenging. However, the formulation we propose in this paper directly applies to the case of multiple textual descriptions per class. Similar to the visual domain, the raw textual descriptions have to go through a feature extraction process, which will be described in Sec 6.2.7. Let us denote the extracted textual feature by $T = \{\mathbf{t}_k \in \mathbb{R}^{d_t}\}_{k=1\cdots N_{sc}}$.

Given a textual description $\mathbf{t}_*$ of a new unseen category, $\mathcal{C}$, the problem can now be defined as predicting a one-vs-all classifier parameters $c(\mathbf{t}_*)$, such that it can be directly used to classify any test image $\mathbf{x}$ as

$$(\mathbf{t}_*)^T \cdot \mathbf{x} > 0 \quad \text{if } \mathbf{x} \text{ belongs to } \mathcal{C}$$

$$(\mathbf{t}_*)^T \cdot \mathbf{x} < 0 \qquad \text{otherwise} \tag{6.1}$$

In what follows, we introduce two possible frameworks for this problem and discuss potential

limitations for them, which leads next to the proposed formulation.

6.2.4 Regression Models

A straightforward way to solve this problem is to pose it as a regression problem where the goal is to use the textual data and the learned classifiers, $\{(\mathbf{t}_k, \mathbf{c}_k)\}_{k=1\cdots N_{sc}}$ to learn a regression function from the textual feature domain to the visual classifier domain, *i.e.* , a function $c(\cdot) : \mathbb{R}^{d_t} \rightarrow \mathbb{R}^{d_v}$. The question is which regression model would be suitable for this problem? and would posing the problem this way give reasonable results?

A typical regression model, such as ridge regression [75] or Gaussian Process (GP) Regression [129], learns the regressor to each dimension of the output domain (the parameters of a linear classifier) separately, *i.e.* a set of function $c^j(\cdot) : \mathbb{R}^{d_t} \rightarrow \mathbb{R}$. Clearly this will not capture the correlation between the visual and textual domain. Instead, a structured prediction regressor would be more suitable since it would learn the correlation between the input and output domain. However, even a structure prediction model, will only learn the correlation between the textual and visual domain through the information available in the input-output pairs $(\mathbf{t}_k, \mathbf{c}_k)$. Here the visual domain information is encapsulated in the pre-learned classifiers and prediction does not have access to the original data in the visual domain. Instead we need to directly learn the correlation between the visual and textual domain and use that for prediction.

Another fundamental problem that a regressor would face, is the sparsity of the data; the data points are the textual description-classifier pairs, and typically the number of classes can be very small compared to the dimension of the classifier space (*i.e.* $N_{sc} \ll d_v$). In a setting like that, any regression model is bound to suffer from an under fitting problem. This can be best explained in terms of GP regression, where the predictive variance increases in the regions of the input space where there are no data points. This will result in poor prediction of classifiers at these regions.

6.2.5 Knowledge Transfer Models

An alternative formulation is to pose the problem as domain adaptation from the textual to the visual domain. In the computer vision context, domain adaptation work has focused on

transferring categories learned from a source domain, with a given distribution of images, to a target domain with different distribution, *e.g.* , images or videos from different sources [177, 136, 90, 43]. What we need is an approach that learns the correlation between the textual domain features and the visual domain features, and uses that correlation to predict new visual classifier given textual features.

In particular, in [90] an approach for learning cross domain transformation was introduced. In that work a regularized asymmetric transformation between points in two domains were learned. The approach was applied to transfer learned categories between different data distributions, both in the visual domain. A particular attractive characteristic of [90], over other domain adaptation models, is that the source and target domains do not have to share the same feature spaces or the same dimensionality.

Inspired by [90], we can formulate the zero-shot learning problem as a domain adaptation. This can be achieve by learning a linear (or nonlinear kernalized) transfer function $\mathbf{W}$ between $\mathcal{T}$ and $\mathcal{V}$. The transformation matrix $\mathbf{W}$ can be learned by optimizing, with a suitable regularizer, over constraints of the form $\mathbf{t}^T\mathbf{W}\mathbf{x} \geq l$ if $\mathbf{t} \in \mathcal{T}$ and $\mathbf{x} \in \mathcal{V}$ belong to the same class, and $\mathbf{t}^T\mathbf{W}\mathbf{x} \leq u$ otherwise. Here l and u are model parameters. This transfer function acts as a compatibility function between the textual features and visual features, which gives high values if they are from the same class and a low value if they are from different classes.

It is not hard to see that this transfer function can act as a classifier. Given a textual feature $\mathbf{t}^*$ and a test image, represented by $\mathbf{x}$, a classification decision can be obtained by $\mathbf{t}_*^T\mathbf{W}\mathbf{x} \gtrless b$ where b is a decision boundary which can be set to $(l + u)/2$. Hence, our desired predicted classifier in Eq 6.1 can be obtained as $c(\mathbf{t}_*) = \mathbf{t}_*^T\mathbf{W}$ (note that the features vectors are amended with ones). However, since learning $\mathbf{W}$ was done over seen classes only, it is not clear how the predicted classifier $c(\mathbf{t}_*)$ will behave for unseen classes. There is no guarantee that such a classifier will put all the seen data on one side and the new unseen class on the other side of that hyperplane.

6.2.6 Problem Formulation

Objective Function

The proposed formulation aims at predicting the hyperplane parameter $\mathbf{c}$ of a one-vs-all classifier for a new unseen class given a textual description, encoded by $\mathbf{t}$ and knowledge learned at the training phase from seen classes. Fig 6.3 illustrates our solution framework. At the training phase three components are learned:

Classifiers: a set of one-vs-all classifiers $\{\mathbf{c}_k\}$ are learned, one for each seen class.

Probabilistic Regressor: Given $\{(\mathbf{t}_k, \mathbf{c}_k)\}$ a regressor is learned that can be used to give a prior estimate for $p_{reg}(\mathbf{c}|\mathbf{t})$.

Domain Transfer Function: Given T and V a domain transfer function, encoded in the matrix $\mathbf{W}$ is learned, which captures the correlation between the textual and visual domains.

Each of these components contains partial knowledge about the problem. The question is how to combine such knowledge to predict a new classifier given a textual description. The new classifier has to be consistent with the seen classes. The new classifier has to put all the seen instances at one side of the hyperplane, and has to be consistent with the learned domain transfer function. This leads to the following constrained optimization problem

$$\begin{aligned}\hat{c}(\mathbf{t}_*) = & \underset{c,\zeta_i}{argmin}[\mathbf{c}^T\mathbf{c} - \alpha {\mathbf{t}_*}^T\mathbf{W}\mathbf{c} - \beta \ln(p_{reg}(\mathbf{c}|\mathbf{t}_*)) \\ & + \gamma \sum \zeta_i] \\ & s.t. : -(\mathbf{c}^T\mathbf{x}_i) \geq \zeta_i,\ \zeta_i \geq 0,\ \ i = 1 \cdots N \\ & {\mathbf{t}_*}^T\mathbf{W}\mathbf{c} \geq l \\ & \alpha, \beta, \gamma, l : \text{hyperparameters}\end{aligned} \tag{6.2}$$

The first term is a regularizer over the classifier $\mathbf{c}$. The second term enforces that the predicted classifier has high correlation with $\mathbf{t}_*^T\mathbf{W}$. The third term favors a classifier that has high probability given the prediction of the regressor. The constraints $-\mathbf{c}^T\mathbf{x}_i \geq \zeta_i$ enforce all the seen data instances to be at the negative side of the predicted classifier hyperplane with some misclassification allowed through the slack variables ζ_i. The constraint ${\mathbf{t}_*}^T\mathbf{W}\mathbf{c} \geq l$ enforces that

the correlation between the predicted classifier and ${\mathbf{t}_*}^T\mathbf{W}$ is no less than l, this is to enforce a minimum correlation between the text and visual features.

Domain Transfer Function

To learn the domain transfer function $\mathbf{W}$ we adapted the approach in [90] as follows. Let $\mathbf{T}$ be the textual feature data matrix and $\mathbf{X}$ be the visual feature data matrix where each feature vector is amended with a 1. Notice that amending the feature vectors with a 1 is essential in our formulation since we need $\mathbf{t}^T\mathbf{W}$ to act as a classifier. We need to solve the following optimization problem

$$\min_{\mathbf{W}} r(\mathbf{W}) + \lambda \sum_i c_i(\mathbf{T}\mathbf{W}\mathbf{X}^T) \tag{6.3}$$

where c_i's are loss functions over the constraints and $r(\cdot)$ is a matrix regularizer. It was shown in [90], under condition on the regularizer, that the optimal $\mathbf{W}$ in Eq 6.3 can be computed using inner products between data points in each of the domains separately, which results in a kernalized non-linear transfer function; hence its complexity does not depend on the dimensionality of either of the domains. The optimal solution of 6.3 is in the form $\mathbf{W}^* = \mathbf{T}\mathbf{K}_T^{-\frac{1}{2}}\mathbf{L}^*\mathbf{K}_X^{-\frac{1}{2}}\mathbf{X}^T$,where $\mathbf{K}_T = \mathbf{T}\mathbf{T}^T$, $\mathbf{K}_X = \mathbf{X}\mathbf{X}^T$. $\mathbf{L}^*$ is computed by minimizing the following minimization problem

$$\min_L [r(L) + \lambda \sum_p c_p(K_T^{\frac{1}{2}} L K_X^{\frac{1}{2}})] \tag{6.4}$$

where $c_p(\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}) = (max(0, (l - e_i\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}e_j)))^2$ for same class pairs of index i,j, or $= (max(0, (e_i\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}e_j - u)))^2$ otherwise, where e_k is a vector of zeros except a one at the k^{th} element, and $u > l$ (note any appropriate l, u could work. In our case, we used $l = 2$, $u = -2$). We used a Frobenius norm regularizer. This energy is minimized using a second order BFGS quasi-Newton optimizer. Once L is computed W^* is computed using the transformation above.

Probabilistic Regressor

There are different regressors that can be used, however we need a regressor that provide a probabilistic estimate $p_{reg}(\mathbf{c}|(t))$. For the reasons explained in Sec 6.2.3, we also need a structure prediction approach that is able to predict all the dimensions of the classifiers together.

Table 6.4: Comparative Evaluation on the Flowers and Birds

Approach	Flowers Avg AUC (+/- std)	Birds Avg AUC (+/- std)
GPR	0.54 (+/- 0.02)	0.52 (+/- 0.001)
TGP	0.58 (+/- 0.02)	0.61 (+/- 0.02)
DA	0.62(+/- 0.03)	0.59 (+/- 0.01)
Our Approach	**0.68** (+/- 0.01)	**0.62** (+/- 0.02)

For these reasons, we use the Twin Gaussian Process (TPG) [20]. TGP encodes the relations between both the inputs and structured outputs using Gaussian Process priors. This is achieved by minimizing the Kullback-Leibler divergence between the marginal GP of the outputs (i.e. classifiers in our case) and observations (i.e. textual features). The estimated regressor output ($\tilde{c}(\mathbf{t}_*)$) in TGP is given by the solution of the following non-linear optimization problem [20] [1].

$$\begin{aligned}\tilde{c}(\mathbf{t}_*) = \underset{c}{argmin}[K_C(\mathbf{c},\mathbf{c}) - 2k_c(\mathbf{c})^T\mathbf{u} - \eta\log(K_C(\mathbf{c},\mathbf{c}) \\ - k_c(\mathbf{c})^T(\mathbf{K}_C + \lambda_c\mathbf{I})^{-1}k_c(c))]\end{aligned} \tag{6.5}$$

where $\mathbf{u} = (\mathbf{K}_T + \lambda_t\mathbf{I})^{-1}k_t(\mathbf{t}_*)$, $\eta = K_T(\mathbf{t}_*,\mathbf{t}_*) - k(\mathbf{t}_*)^T\mathbf{u}$, $K_T(\mathbf{t}_l,\mathbf{t}_m)$ and $K_C(\mathbf{c}_l,\mathbf{c}_m)$ are Gaussian kernel for input feature $\mathbf{t}$ and output vector $\mathbf{c}$. $k_c(\mathbf{c}) = [K_C(\mathbf{c},\mathbf{c}_1),\cdots,K_C(\mathbf{c},\mathbf{c}_{N_{sc}})]^T$. $k_t(\mathbf{t}_*) = [K_T(\mathbf{t}_*,\mathbf{t}_1),\cdots,K_T(\mathbf{t}_*,\mathbf{t}_{N_{sc})}]^T$. λ_t and λ_c are regularization parameters to avoid overfitting. This optimization problem can be solved using a second order, BFGS quasi-Newton optimizer with cubic polynomial line search for optimal step size selection [20]. In this case the classifier dimension are predicted jointly. In this case $p_{reg}(\mathbf{c}|\mathbf{t}_*)$ is defined as a normal distribution.

$$p_{reg}(\mathbf{c}|\mathbf{t}_*) = \mathcal{N}(\mu_c = \tilde{c}(\mathbf{t}_*), \Sigma_c = \mathbf{I}) \tag{6.6}$$

The reason that $\Sigma_c = \mathbf{I}$ is that TGP does not provide predictive variance, unlike Gaussian Process Regression. However, it has the advantage of handling the dependency between the dimensions of the classifiers $\mathbf{c}$ given the textual features $\mathbf{t}$.

[1] notice we are using $\tilde{\mathbf{c}}$ to denote the output of the regressor, while using $\hat{\mathbf{c}}$ to denote the output of the final optimization problem in Eq 6.2

Solving for $\hat{c}$ as a quadratic program

According to the definition of $p_{reg}(\mathbf{c}|\mathbf{t}_*)$ for TGP, $\ln p(\mathbf{c}|\mathbf{t}_*)$ is a quadratic term in c in the form

$$-\ln p(\mathbf{c}\mathbf{t}_*) \propto (\mathbf{c}-\tilde{c}(\mathbf{t}_*))^{\mathbf{T}}(\mathbf{c}-\tilde{c}(\mathbf{t}_*)) = \mathbf{c}^{\mathbf{T}}\mathbf{c} - 2\mathbf{c}^{\mathbf{T}}\tilde{c}(\mathbf{t}_*) + \tilde{c}(\mathbf{t}_*)^{\mathbf{T}}\tilde{c}(\mathbf{t}_*) \tag{6.7}$$

We reduce $-\ln p(\mathbf{c}|\mathbf{t}_*)$ to $-2\mathbf{c}^T\tilde{c}(\mathbf{t}_*))$, since 1) $\tilde{c}(\mathbf{t}_*)^T\tilde{c}(\mathbf{t}_*)$ is a constant (*i.e.* does not affect the optimization), 2) $\mathbf{c}^T\mathbf{c}$ is already included as regularizer in equation 6.2. In our setting, the dot product is a better similarity measure between two hyperplanes. Hence, $-2\mathbf{c}^T\tilde{c}(\mathbf{t}_*)$ is minimized. Given $-\ln p(\mathbf{c}|\mathbf{t}_*)$ from the TGP and $\mathbf{W}$, Eq 6.2 reduces to a quadratic program on $\mathbf{c}$ with linear constraints. We tried different quadratic solvers, however the IBM CPLEX solver gives the best performance in speed and optimization for our problem.

6.2.7 Experiments

6.2.8 Datasets

We used the CU200 Birds [173] (200 classes - 6033 images) and the Oxford Flower-102 [114] (102 classes - 8189 images) image dataset to test our approach, since they are among the largest and widely used fine-grained datasets. We generate textual descriptions for each class in both datasets. The CU200 Birds image dataset was created based on birds that have a corresponding Wikipedia article, so we have developed a tool to automatically extract Wikipedia articles given the class name. The tool succeeded to automatically generate 178 articles, and the remaining 22 articles was extracted manually from Wikipedia. These mismatches happens only when article title is a different synonym of the same bird class. On the other hand, Flower image dataset was not created using the same criteria as the Bird dataset, so classes of the Flower dataset classes does not necessarily have corresponding Wikipedia article. The tool managed to generate about 16 classes from Wikipedia out of 102, the remaining 86 articles was generated manually for each class from Wikipedia, Plant Database [2], Plant Encyclopedia [3], and BBC articles [4]. We plan to make the extracted textual description available as augmentations of these datasets.

[2] http://plants.usda.gov/java/

[3] http://www.theplantencyclopedia.org/wiki/Main_Page

[4] http://www.bbc.co.uk/science/0/

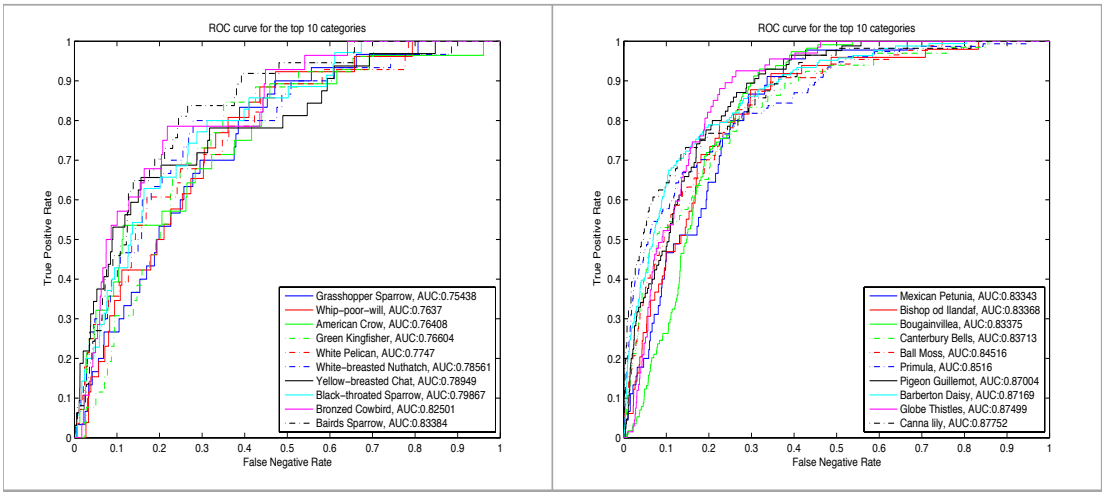

Figure 6.4: **Left**: ROC curves of best 10 predicted classes (best seen in color) for Bird datasets respectively. **Right**: ROC curves of best 10 predicted classes (best seen in color) for Flower datasets respectively.

Sample textual description can be found in the supplementary material.

6.2.9 Extracting Textual Features

The textual features were extracted in two phases, which are typical in document retrieval literature. The first phase is an indexing phase that generates textual features with tf-idf (Term Frequency-Inverse Document Frequency) configuration (Term frequency as local weighting while inverse document frequency as a global weighting). The tf-idf is a measure of how important is a word to a text corpus. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others. We used the normalized frequency of a term in the given textual description [147]. The inverse document frequency is a measure of whether the term is common; in this work we used the standard logarithmic idf [147]. The second phase is a dimensionality reduction step, in which Clustered Latent Semantic Indexing (CLSI) algorithm [181] is used. CLSI is a low-rank approximation approach for dimensionality reduction, used for document retrieval. In the Flower Dataset, tf-idf features $\in \mathbb{R}^{8875}$ and after CLSI the final textual features $\in \mathbb{R}^{102}$. In the Birds Dataset, tf-idf features is in $\mathbb{R}^{7086}$ and after CLSI the final textual features is in $\mathbb{R}^{200}$.

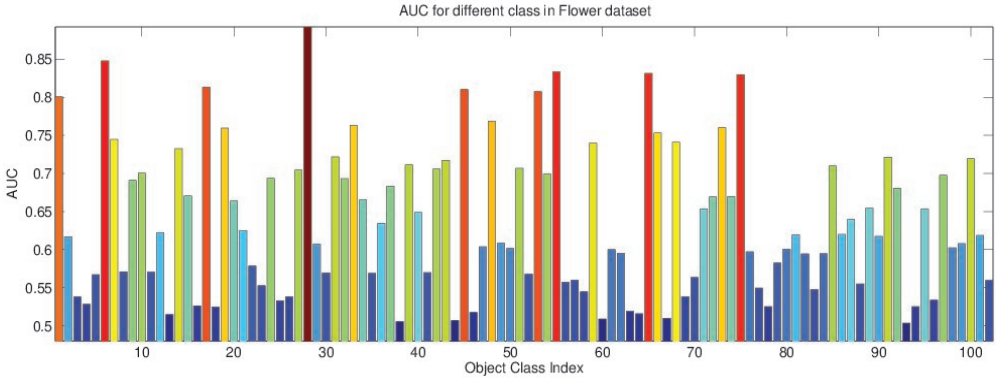

Figure 6.5: AUC of the predicated classifiers for all classes of the flower datasets

6.2.10 Visual features

We used the Classeme features [15] as the visual feature for our experiments since they provide an intermediate semantic representation of the input image. Classeme features are output of a set of classifiers corresponding to a set of C category labels, which are drawn from an appropriate term list defined in [15], and not related to our textual features. For each category $c \in \{1 \cdots C\}$, a set of training images is gathered by issuing a query on the category label to an image search engine. After a set of coarse feature descriptors (Pyramid HOG, GIST, *etc.*) is extracted, a subset of feature dimensions was selected [15], and a one-versus-all classifier ϕ_c is trained for each category. The classifier output is real-valued, and is such that $\phi_c(x) > \phi_c(y)$ implies that x is more similar to class c than y is. Given an image x, the feature vector (descriptor) used to represent it is the classeme vector $[\phi_1(x), \cdots, \phi_C(x)]$. The Classeme feature is of dimensionality 2569.

6.2.11 Experimental Results

Evaluation Methodology and Metrics: Similar to zero-shot learning literature, we evaluated the performance of an unseen classifier in a one-vs-all setting where the test images of unseen classes are considered to be the positives and the test images from the seen classes are considered to be the negatives. We computed the ROC curve and report the area under that curve (AUC) as a comparative measure of different approaches. In zero-shot learning setting the test

Table 6.5: Percentage of classes that the proposed approach makes an improvement in predicting over the baselines (relative to the total number of classes in each dataset

baseline	Flowers (102) % improvement	Birds (200) % improvement
GPR	100 %	98.31 %
TGP	66 %	51.81 %
DA	54%	56.5%

data from the seen class are typically very large compared to those from unseen classes. This makes other measures, such as accuracy, useless since high accuracy can be obtained even if all the unseen class test data are wrongly classified; hence we used ROC curves, which are independent of this problem. Five-fold cross validation over the classes were performed, where in each fold 4/5 of the classes are considered as "seen classes" and are used for training and 1/5th of the classes were considered as "unseen classes" where their classifiers are predicted and tested. Within each of these class-folds, the data of the seen classes are further split into training and test sets. The hyper-parameters for the approach were selected through another five-fold cross validation within the class-folds (i.e. the 80% training classes are further split into 5 folds to select the hyper-parameters).

Baselines: Since our work is the first to predict classifiers based on pure textual description, there are no other reported results to compare against. However, we designed three state-of-the-art baselines to compare against, which are designed to be inline with our argument in Sec 6.2.3. Namely we used: 1) A Gaussian Process Regressor (GPR) [129], 2) Twin Gaussian Process (TGP) [20] as a structured regression method, 3) Nonlinear Asymmetric Domain Adaptation (DA) [90]. The TGP and DA baselines are of particular importance since our formulation utilizes them, so we need to test if the formulation is making any improvement over them. It has to be noted that we also evaluate TGP and DA as alternative formulations that we are proposing for the problem, none of them was used in the same context before.

Results: Table 6.4 shows the average AUCs for the proposed approach in comparison to the three baselines on both datasets. GPR performed poorly in all classes in both data sets, which was expected since it is not a structure prediction approach. The DA formulation outperformed TGP in the flower dataset but slightly underperformed on the Bird dataset. The proposed approach outperformed all the baselines on both datasets, with significant difference on the flower

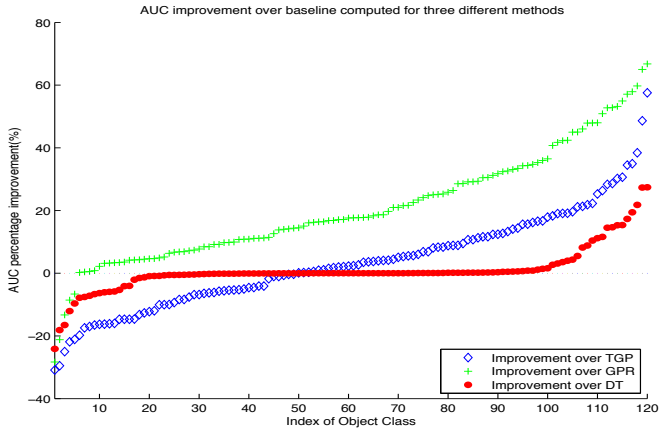

Figure 6.6: AUC improvement over the three baselines on Flower dataset. The improvement is sorted in an increasing order for each baseline separately

dataset. It is also clear that the TGP performance was improved on the Bird dataset since it has more classes (more points are used for prediction).

Figure 6.4 shows the ROC curves for our approach on best predicted unseen classes from the Birds dataset on the Left and Flower dataset on the right. Figure 6.5 shows the AUC for all the classes in the Flower dataset.

Figure 6.2.11, on the right, shows the improvement over the three baseline for each class, where the improvement is calculated as (our AUC- baseline AUC)/ baseline AUC %. Table 6.5 shows the percentage of the classes which our approach makes a prediction improvement for each of the three baselines.Table 6.6 shows the five classes in Flower dataset where our approach made the best average improvement.

The point of that table is to show that in these cases both TGP and DA did poorly while our formulation that is based on both of them did significantly better. This shows that our formulation does not simply combine the best of the two approaches but can significantly improve the prediction performance.

To evaluate the effect of the constraints in the objective function, we removed the constraints $-(\mathbf{c}^T\mathbf{x}_i) \geq \zeta_i$ which try to enforces all the seen examples to be on the negative side of the predicted classifier hyperplane and evaluated the approach. The result on the flower dataset (using one fold) was reduced to average AUC=0.59 compared to AUC=0.65 with the

Table 6.6: Top-5 classes with highest combined improvement in Flower dataset

class	TGP (AUC)	DA (AUC)	Our (AUC)	% Improv.
2	0.51	0.55	0.83	57%
28	0.52	0.54	0.76	43.5%
26	0.54	0.53	0.76	41.7%
81	0.52	0.82	0.87	37%
37	0.72	0.53	0.83	35.7 %

constraints. Similarly, we evaluated the effect of the constraint $\mathbf{t}_*^T\mathbf{W}\mathbf{c} \geq l$. The result was reduced to average AUC=0.58 compared to AUC=0.65 with the constraint. This illustrates the importance of this constraint in the formulation.

6.2.12 Conclusion and Future Work

We explored the problem of predicting visual classifiers from textual description of classes with no training images. We investigated and experimented with different formulations for the problem within the fine-grained categorization context. We proposed a novel formulation that captures information between the visual and textual domains by involving knowledge transfer from textual features to visual features, which indirectly leads to predicting the visual classifier described by the text. In the future, we are planning to propose a kernel version to tackle the problem instead of using linear classifiers. Furthermore, we will study predicting classifiers from complex-structured textual features.

Chapter 7

Conclusion and Future Work

In this thesis we presented results of our investigation on the subject of abnormality in images. We explored definition of typicality and abnormality in images based on human visual understanding, along with challenges of detecting abnormalities automatically (chapter 1). In chapter 2, We made the biggest dataset of abnormal images and conducted a large-scale human subject experiment to investigate how humans think about abnormal images. We proposed a diverse list of abnormality reasons by human responses, and inferred a taxonomy of visual cues that make an image abnormal.

We also introduced a model to predict abnormality of objects by reasonings in terms of attributes. We show improvements over standard baselines on abnormality prediction. With such a predictions our model can also report its reasoning in terms of abnormal attributes (chapter 3). In the fourth chapter of this thesis, we approached the challenging research question: what make an image look abnormal? Based on three major components of the inferred taxonomy in chapter 2, we built computer vision models that can detect an abnormal images and reason about this decision in terms of three surprise scores.

We conducted a study on the generalization capacity of convolution neural networks in chapter 5. There are several points that we can conclude from this study. The state-of-the-art CNN object classifiers fail drastically when they are applied on atypical images. Atypicality is not necessarily equivalent to samples on the boundary, which common loss functions try to emphasize in learning. However, atypical images show extreme changes in visual features, which are still understandable to the human visual system. The main result of this chapter is that involving information about the typicality/atypicality of training samples as a weighting term in the loss function helps greatly in enhancing the performance on unseen atypical examples, when training only using typical examples.

Additionally, we presented (in chapter 6) an end to end framework for detection, classification and description of fine grained object categories based on localized attributes. Since our model is based on Deformable Part Model, it is able to handle deformation in the shape of the objects as well as diverse object poses. In addition to detecting the objects we localize parts of the objects and describe them in terms of visual attributes. These attribute are part based and specifically designed for one part. Having these attributes we are able to describe objects in addition to classify them. As visual attributes are meaningful for human, we can learn a new category only by its textual description. Also, we explored the problem of predicting visual classifiers from textual description of classes with no training images. We proposed a novel formulation that captures information between the visual and textual domains by involving knowledge transfer from textual features to visual features, which indirectly leads to predicting the visual classifier described by the text.

Appendices

Appendix A

Visual Analysis of Fine Art

In this chapter we study some challenging problems in the field of visual analysis of fine art paintings. Although paintings carry rich amount of information through the medium of images, but they cannot be analyzed by straightforward applications of state-of-the-art computer vision algorithms. This is mainly because paintings do not necessarily represent a realistic scene, which follow physical rules of the real world. Additionally, paintings as artworks, are made by incorporating the notion of creativity in addition to techniques and materials that are used.

In this chapter we are interested to find influence path between artworks and artists based on visual analysis of their artworks. We also find paintings that are meaningfully different from all other paintings (abnormal in respect to some styles/genres), and quantify creativity of paintings as being influential and novel. Toward these ambitious goals, we design a framework to learn a powerful representation for images of paintings. These representation is based on applying metric learning on an extensive set of raw visual features. Later in this chapter, we use this visual representation to compare paintings and find influence paths between artists and quantify creativity of artworks.

A.1 A Unified Framework For Painting Classification

1. Babak Saleh, Ahmed Elgammal: Large-scale Classification of Fine-Art Paintings: Learning The Right Metric on The Right Feature, Journal of Digital Art History, Oct 2016.

2. Babak Saleh, Ahmed Elgammal: A Unified Framework for Painting Classification, ICDMW 2015.

In the past few years, the number of fine-art collections that are digitized and publicly available has been growing rapidly. With the availability of such large collections of digitized artworks

comes the need to develop multimedia systems to archive and retrieve this pool of data. Measuring the visual similarity between artistic items is an essential step for such multimedia systems, which can benefit more high-level multimedia tasks. In order to model this similarity between paintings, we should extract the appropriate visual features for paintings and find out the best approach to learn the similarity metric based on these features. We investigate a comprehensive list of visual features and metric learning approaches to learn an optimized similarity measure between paintings. We develop a machine that is able to make aesthetic-related semantic-level judgments, such as predicting a painting's style, genre, and artist, as well as providing similarity measures optimized based on the knowledge available in the domain of art historical interpretation. Our experiments show the value of using this similarity measure for the aforementioned prediction tasks.

A.1.1 Introduction

In the past few years, the number of fine-art collections that are digitized and publicly available has been growing rapidly. Such collections span classical [1] and modern and contemporary artworks [2]. With the availability of such large collections of digitized artworks comes the need to develop multimedia systems to archive and retrieve this pool of data. Typically these collections, in particular early modern ones, come with metadata in the form of annotations by art historians and curators, including information about each painting's artist, style, date, genre, etc. For online galleries displaying contemporary artwork, there is a need to develop automated recommendation systems that can retrieve "similar" paintings that the user might like to buy. This highlights the need to investigate metrics of visual similarity among digitized paintings that are optimized for the domain of painting.

The field of computer vision has made significant leaps in getting digital systems to recognize and categorize objects and scenes in images and videos. These advances have been driven by a wide spread need for the technology, since cameras are everywhere now. However a person looking at a painting can make sophisticated inferences beyond just recognizing a tree, a chair,

[1] Examples: Wikiart; Arkyves; BBC Yourpainting

[2] Examples: Behance; Artfinder ;Artsy; Artnet

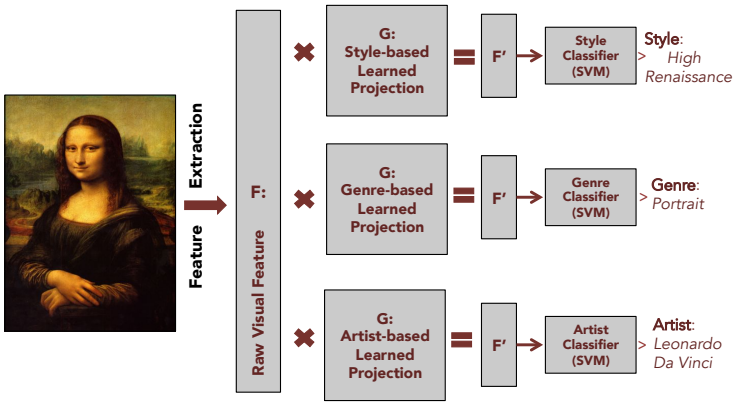

Figure A.1: Illustration of our system for classification of fine-art paintings. We investigated variety of visual features and metric learning approaches to recognize *Style, Genre* and *Artist* of a painting.

or the figure of Christ. Even individuals without specific art historical training can make assumptions about a painting's genre (portrait or landscape), its style (impressionist or abstract), what century it was created, the artists who likely created the work and so on. Obviously, the accuracy of such assumptions depends on the viewer's level of knowledge and exposure to art history. Learning and judging such complex visual concepts is an impressive ability of human perception [4].

The ultimate goal of our research is to develop a machine that is able to make aesthetic-related semantic-level judgments, such as predicting a painting's style, genre, and artist, as well as providing similarity measures optimized based on the knowledge available in the domain of art historical interpretation. Immediate questions that arise include, but are not limited to: What visual features should be used to encode information in images of paintings? How does one weigh different visual features to achieve a useful similarity measure? What type of art historical knowledge should be used to optimize such similarity measures? In this chapter we address these questions and aim to provide answers that can benefit researchers in the area of computer-based analysis of art. Our work is based on a systematic methodology and a comprehensive evaluation on one of the largest available digitized art datasets.

Artists use different concepts to describe paintings. In particular, stylistic elements, such as space, texture, form, shape, color, tone and line are used. Other principles include movement,

Task Name	List of Members
Style	Abstract Expressionism(1); Action Painting(2); Analytical Cubism(3); Art Nouveau-Modern Art(4); Baroque(5); Color Field Painting(6); Contemporary Realism(7); Cubism(8); Early Renaissance(9); Expressionism(10); Fauvism(11); High Renaissance(12); Impressionism(13); Mannerism-Late-Renaissance(14); Minimalism(15); Primitivism-Naive Art(16); New Realism(17); Northern Renaissance(18); Pointillism(19); Pop Art(20); Post Impressionism(21); Realism(22); Rococo(23); Romanticism(24); Symbolism(25); Synthetic Cubism(26); Ukiyo-e(27)
Genre	Abstract painting(1); Cityscape(2); Genre painting(3); Illustration(4); Landscape(5); Nude painting(6); Portrait(7); Religious painting(8); Sketch and Study(9); Still Life(10)
Artist	Albrecht Durer(1); Boris Kustodiev(2); Camille Pissarro(3); Childe Hassam(4); Claude Monet(5); Edgar Degas(6); Eugene Boudin(7); Gustave Dore(8); Ilya Repin(9); Ivan Aivazovsky(10); Ivan Shishkin(11); John Singer Sargent(12); Marc Chagall(13); Martiros Saryan(14); Nicholas Roerich(15); Pablo Picasso(16); Paul Cezanne(17); Pierre-Auguste Renoir(18); Pyotr Konchalovsky(19); Raphael Kirchner(20); Rembrandt(21); Salvador Dali(22); Vincent van Gogh(23)

Table A.1: List of Styles, Genres and Artists in our collection of fine-art paintings. Numbers in the parenthesis are index of the row/column in confusion matrices A.5, A.6& A.7 accordingly.

unity, harmony, variety, balance, contrast, proportion, and pattern. To this might be added physical attributes, like brush strokes as well as subject matter and other descriptive concepts [60].

For the task of computer analyses of art, researchers have engineered and investigated various visual features[3] that encode some of these artistic concepts, in particular brush strokes and color, which are encoded as low-level features such as texture statistics and color histograms (e.g. [99, 100]). Color and texture are highly prone to variations during the digitization of paintings; color is also affected by a painting's age. The effect of digitization on the computational analysis of paintings is investigated in great depth by Polatkan et al. [127]. This highlights the need to carefully design visual features that are suitable for the analysis of paintings.

Clearly, it would be a cumbersome process to engineer visual features that encode all the aforementioned artistic concepts. Recent advances in computer vision, using deep neural networks, showed the advantage of "learning" the features from data instead of engineering such

[3]In contrast to art disciplines, in the fields of computer vision and machine learning, researchers use the term"visual features" to denote statistical measurements that are extracted from images for the task of classification. In this chapter we stick to this typical terminology.

features. However, It would also be impractical to learn visual features that encode such artistic concepts, since that would require extensive annotation of these concepts in each image within a large training and testing dataset. Obtaining such annotations require expertise in the field of art history that can not be achieved with typical crowed-sourcing annotators.

Given the aforementioned challenges to engineering or learning suitable visual features for painting, in this chapter we follow an alternative strategy. We mainly investigate different state-of-the-art visual elements, ranging from low-level elements to semantic-level elements. We then use metric learning to achieve optimal similarity metrics between paintings that are optimized for specific prediction tasks, namely style, genre, and artist classification. We chose these tasks to optimize and evaluate the metrics since, ultimately, the goal of any art recommendation system would be to retrieve artworks that are similar along the directions of these high-level semantic concepts. Moreover, annotations for these tasks are widely available and more often agreed-upon by art historians and critics, which facilitates training and testing the metrics.

In this chapter we investigate a large space of visual features and learning methodologies for the aforementioned prediction tasks. We propose and compare three learning methodologies to optimize such tasks. We present results of a comprehensive comparative study that spans four state-of-the-art visual features, five metric learning approaches and the proposed three learning methodologies, evaluated on the aforementioned three artistic prediction tasks.

A.1.2 Related Work

On the subject of painting, computers have been used for a diverse set of tasks. Traditionally, image processing techniques have been used to provide art historians with quantification tools, such as pigmentation analysis, statistical quantification of brush strokes, etc. We refer the reader to [156, 13] for comprehensive surveys on this subject.

Several studies have addressed the question of which features should be used to encode information in paintings. Most of the research concerning the classification of paintings utilizes low-level features encoding color, shadow, texture, and edges. For example Lombardi [100] has presented a study of the performance of these types of features for the task of artist classification among a small set of artists using several supervised and unsupervised learning methodologies.

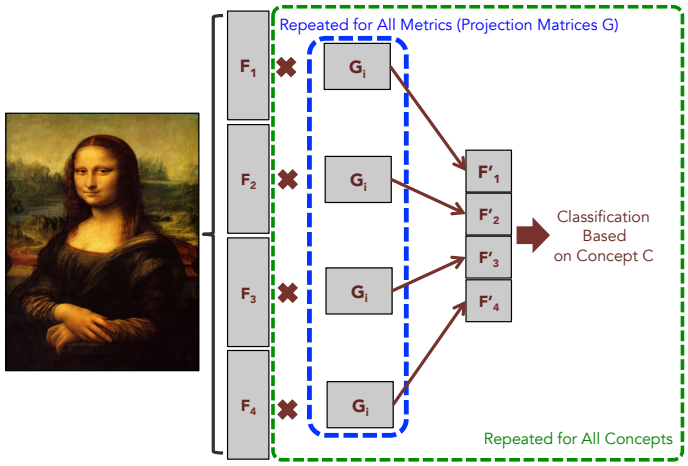

Figure A.2: Illustration of our second methodology - Feature Fusion.

In that paper the style of the painting was identified as a result of recognizing the artist.

Since brushstrokes provide a signature that can help identify the artist, designing visual features that encode brushstrokes has been widely adapted.(e.g. [128, 98, 101, 81, 14, 99]). Typically, texture statistics are used for that purpose. However, as mentioned earlier, texture features are highly affected by the digitization resolution. Researchers also investigated the use of features based on local edge orientation histograms, such as SIFT [166] and HOG [36]. For example, [51] used SIFT features within a Bag-of-words pipeline to discriminate among a set of eight artists.

Arora et al. [5] presented a comparative study for the task of style classification, which evaluated low-level features, such as SIFT and Color SIFT [166], versus semantic-level features, namely Classemes [165], which encodes object presence in the image. It was found that semantic-level features significantly outperform low-level features for this task. However the evaluation was conducted on a small dataset of 7 styles, with 70 paintings in each style. Carneiro et al [29] also concluded that low-level texture and color features are not effective because of inconsistent color and texture patterns that describe the visual classes in paintings.

More recently, Saleh et al [140] used metric learning approaches for finding influence paths between painters based on their paintings. They evaluated three metric learning approaches to optimize a metric over low-level HOG features. In contrast to that work, the evaluation presented in this chapter is much wider in scope since we address three tasks (style, genre and

artist prediction), we cover features spanning from low-level to semantic-level and we evaluate five metric learning approaches. Moreover, The dataset of [140] has only 1710 images from 66 artists, while we conducted our experiments on 81,449 images painted by 1119 artists. Bar et al [8] proposed an approach for style classification based on features obtained from a convolution neural network pre-trained on an image categorization task. In contrast we show that we can achieve better results with much lower dimensional features that are directly optimized for style and genre classification. Lower dimensionality of the features is preferred for indexing large image collections.

A.1.3 Methodology

In this section we explain the methodology that we follow to find the most appropriate combination of visual features and metrics that produce accurate similarity measurements. We acquire these measurements to mimic the art historian's ability to categorize paintings based on their style, genre and the artist who made it. In the first step, we extract visual features from the image. These visual features range from low-level (e.g. edges) to high-level (e.g. objects in the painting). More importantly, in the next step we learn how to adjust these features for different classification tasks by learning the appropriate metrics. Given the learned metric we are able to project paintings from a high dimensional space of raw visual information to a meaningful space with much lower dimensionality. Additionally, learning a classifier in this low-dimensional space can be easily scaled up for large collections.

In the rest of this section: First, we introduce our collection of fine-art paintings and explain what are the tasks that we target in this work. Later, we explore methodologies that we consider in this work to find the most accurate system for aforementioned tasks. Finally, we explain different types of visual features that we use to represent images of paintings and discuss metric learning approaches that we applied to find the proper notion of similarity between paintings.

A.1.4 Dataset and Proposed Tasks

In order to gather our collection of fine-art paintings, we used the publicly available dataset of *"Wikiart paintings"*[4]; which, to the best of our knowledge, is the largest online public collection of digitized artworks. This collection has images of 81,449 fine-art paintings from 1,119 artists ranging from fifteen centuries to contemporary artists. These paintings are from 27 different styles (Abstract, Byzantine, Baroque, etc.) and 45 different genres (Interior, Landscape, etc.) Previous work [140, 29] used different resources and made smaller collections with limited variability in terms of style, genre and artists. The work of [8] is the closest to our work in terms of data collection procedure, but the number of images in their collection is half of ours.

We target automatic classification of paintings based on their style, genre and artist using visual features that are automatically extracted using computer vision algorithms. Each of these tasks has its own challenges and limitations. For example, there are large variations in terms of visual appearances in paintings from one specific style. However, this variation is much more limited for paintings by one artist. These larger intra-class variations suggests that style classification based on visual features is more challenging than artist classification. For each of the tasks we selected a subset of the data that ensure enough samples for training and testing. In particular for style classification we use a subset of the date with 27 styles where each style has at least 1500 paintings with no restriction on genre or artists, with a total of 78,449 images. For genre classification we use a subset with 10 genre classes, where each genre has at least 1500 paintings with no restriction of style or genre, with a total of 63,691 images. Similarly for artist classification we use a subset of 23 artists, where each of them has at least 500 paintings, with a total of 18,599 images. Table A.1 lists the set of style, genre, and artist labels.

A.1.5 Classification Formulations

In order to classify paintings based on their style, genre or artist we followed three methodologies.

Metric Learning: First, as depicted in figure A.1, we extract visual features from images of paintings. For each of these prediction tasks, we learn a similarity metric optimized for it,

[4]http://www.wikiart.org/

i.e. style-optimized metric, genre-optimized metric and artist-optimized metric. Each metric induces a projector to a corresponding feature space optimized for the corresponding task. Having the metric learned, we project the raw visual features into the new optimized feature space and learn classifiers for the corresponding prediction task. For that purpose we learn a set of one-vs-all SVM classifiers for each of the labels in table A.1 for each of the tasks.

While our first strategy focuses on classification based on combinations of a metric and a visual feature, the next two methodologies that we followed fuse different features or different metrics.

Feature fusion: The second methodology that we used for classification is depicted in figure A.2. In this case, we extract different types of visual features (four types of features as will explained next). Based on the prediction task (e.g. style) we learn the metric for each type of feature as before. After projecting these features separately, we concatenate them to make the final feature vector. The classification will be based on training classifiers using these final features. This feature fusion is important as we want to capture different types of visual information by using different types of features. Also concatenating all features together and learn a metric on top of this huge feature vector is computationally intractable. Because of this issue, we learn metrics on feature separately and after projecting features by these metrics, we can concatenate them for classification purposes.

Metric-fusion: The third methodology (figure A.3) projects each visual features using multiple metrics (in our experiment we used five metrics as will be explained next) and then fuses the resulting optimized feature spaces to obtain a final feature vector for classification. This is an important strategy, because each one of the metric learning approaches use a different criteria to learn the similarity measurement. By learning all metrics individually (on the same type of feature), we make sure that we took into account all criteria (e.g. information theory along with neighbor hood analysis).

Raw Visual Features

Visual features in computer vision literature are either engineered and extracted in an unsupervised way (e.g. HOG, GIST) or learned based on optimizing a specific task, typically categorization of objects or scenes (e.g. CNN-based features). This results in high-dimensional

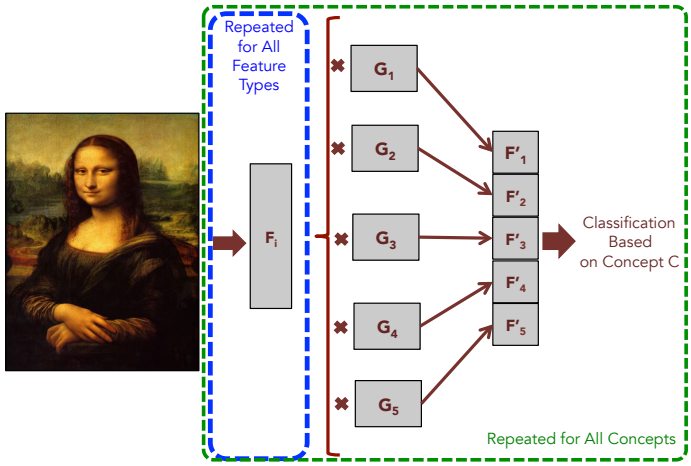

Figure A.3: Illustration of our third methodology– Metric Fusion.

feature vectors that might not necessary correspond to nameable (semantic-level) characteristics of an image. Based on the ability to find a meaning, visual features can be categorized into low-level and high-level. Low-level features are visual descriptors that there is no explicit meaning for each dimension of them, while high-level visual features are designed to capture some notions (usually objects). For this work, we investigated some state-of-the-art representatives of these two categories:

Low-level Features: On one hand, in order to capture low-level visual information we extracted GIST features [116], which are holistic features that are designed for scene categorization. GIST features provide a 512 real-valued representation that implicitly captures the dominant spatial structure of the image.

Learned Semantic-level Features: On the other hand, for the purpose of semantic representation of the images, we extracted three object-based representation of the images: Classeme [165], Picodes [16], and CNN-based features [89]. In all these three features, each element of the feature vector represents the confidence of the presence of an object-category in the image, therefore they provide a semantic encoding of the images. However, for learning these features, the object-categories are generic and are not art-specific. First two features are designed to capture the presence of a set of basic-level object categories as following: a list of entry-level categories (e.g. horse and cross) is used for downloading a large collection of images from the web. For each image a comprehensive set of low-level visual features are extracted and one classifier is

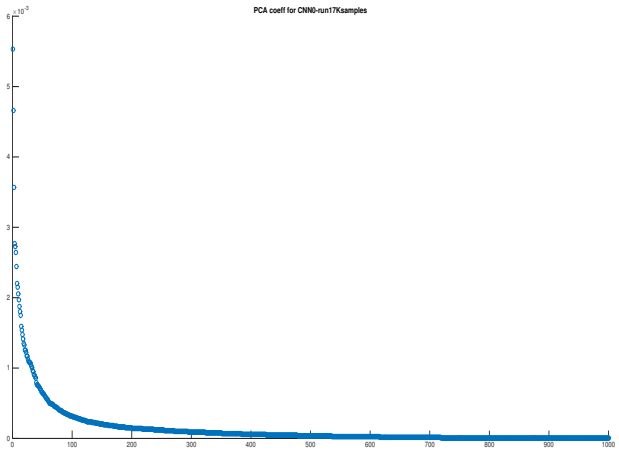

Figure A.4: Plot of PCA coefficients corresponding to CNN features extracted from paintings.

learned for each category. For a given test image, these classifiers are applied on the image and the responses (confidences) make the final feature vector. We followed the implementation of [15] and for each image extracted a 2659 dimensional real-valued Classeme feature vector and a 2048 dimensional binary-value Picodes feature.

Convolutional Neural Networks(CNN) [97] showed a remarkable performance for the task of large-scale image categorization [89]. CNNs have four convolutional layers followed by three fully connected layers. Bar et al [8] showed that a combination of the output of these fully connected layers achieve a superior performance for the task of style classification of paintings. Following this observation we used the last layer of a pre-trained CNN [89] (1000 dimensional real-valued vectors) as another feature vector.

A.1.6 Metric Learning as Feature Projection

The purpose of Metric Learning is to find some pair-wise real-valued function $d_M(x, x')$ which is non-negative, symmetric, obeys the triangle inequality and returns zero if and only if x and x' are the same point. Training such a function in a general form can be seen as the following optimization problem:

$$\min_M l(M, D) + \lambda R(M) \tag{A.1}$$

This optimization has two sides, first it tries to minimize the amount of loss $l(M, D)$ by using metric M over data samples D while trying to adjust the model by the regularization term $R(M)$. The first term shows the accuracy of the trained metric and second one estimates its capability over new data and avoids overfitting. Based on the enforced constraints, the resulted metric can be linear or non-linear and depending on the amount of labels used for training, it can be supervised or unsupervised.

For consistency over the metric learning algorithms, we need to fix the notation first. We learn the matrix M that will be used in Generalized Mahalanobis Distance: $d_M(x, x') = \sqrt{(x - x')'M(x - x')}$, where M by definition is a positive semi-definite matrix and can be decomposed as $M = G^T G$. We use this matrix G to project raw visual features. Measuring similarity in this projection space is simply computing the euclidean distance between two item.

It is interesting that we can reduce the dimension of features during learning the metric when M is a low rank matrix. More importantly, there are significantly important information in the ground truth annotation associated with paintings that we use to learn a more reliable metric in a supervised fashion for both the linear and non-linear cases. We consider following approaches that differ based on the form of M or the amount of regularization: Neighborhood Component Analysis (NCA) [67], Large Margin Nearest Neighbors (LMNN) [171], Boost Metric [150], Information Theory Metric Learning (ITML) [37], Metric Learning for Kernel Regression (MLKR) [170].

A.1.7 Experiments

Visual Features

As we explained in section A.1.3, we extract GIST features as low-level visual features and Classeme, Picodes and CNN-based features as the high-level semantic features. We followed the original implementation of Oliva and Torralba [116] to get a 512 dimensional feature vector. For Classeme and Picodes we used the implementation of Bergamo et al [165], resulting in 2659 dimensional Classeme features and 2048 dimensional Picodes features. We used the implementation of Vedaldi and Lenc [167] to extract 1000 dimensional feature vectors of the

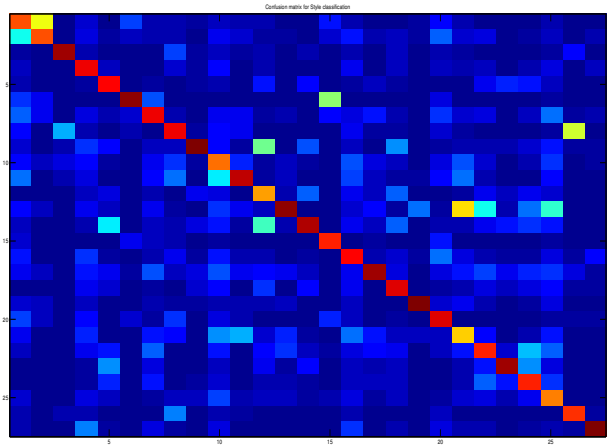

Figure A.5: Confusion matrix for Style classification. Confusions are meaningful only when seen in color.

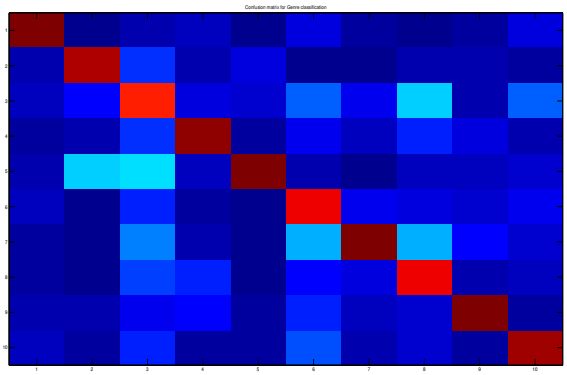

Figure A.6: Confusion matrix for Genre classification. Confusions are meaningful only when seen in color.

last layer of CNN.

Object-based representations of the images produce feature vectors that are much higher in dimensionality than GIST descriptors. In the sake of a fair comparison of all types of features for the task of metric learning, we transformed all feature vectors to have the same size as GIST (512 dimensional). We did this by applying Principle Component Analysis (PCA) for each type and projecting the original features onto the first 512 eigenvectors (with biggest eigenvalues). In order to verify the quality of projection, we looked at the corresponding coefficients of eigenvalues for PCA projections. Independent of feature type, the value of these

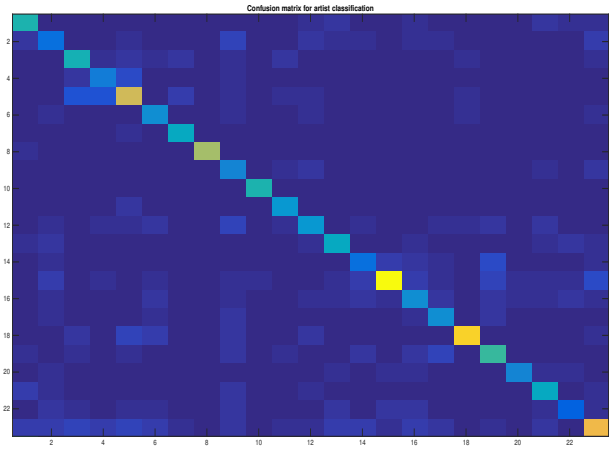

Figure A.7: Confusion matrix for Artist classification. Confusions are meaningful only when seen in color.

Metric / Features	GIST	Classemes	Picodes	CNN	Dim.
Baseline	10.83	22.62	20.76	12.32	512
Boost	**16.07**	**31.77**	**28.58**	15.18	512
ITML	13.02	30.67	28.42	15.34	512
LMNN	12.54	27	24.14	**16.83**	100
MLKR	12.65	24.12	14.86	12.63	512
NCA	13.29	28.19	24.84	16.37	27

Table A.2: Accuracy for the task of style classification

coefficients drops significantly after the first 500 eigenvectors. For example, figure A.4 plots these coefficients of PCA projection for CNN features. Summation of the first 500 coefficients is 95.88% of the total summation. This shows that our projections (with 512 eigenvectors) captures the true underlying space of the original features. Using these reduced features speeds up the metric learning process as well.

Metric Learning

We used implementation of [171] to learn LMNN metric(both version of linear and non-linear) and MLKR [5]. For the BoostMetric we slightly adjusted the implementation of [150]. For NCA

[5] http://www.cse.wustl.edu/ kilian/index.html

we adopted its implementation by Fowlkes[6] to work on large scale feature vectors smoothly. For the case of ITML metric learning, we followed the original implementation of authors with the default setting. For the rest of methods, parameters are chosen through a grid search that finds the minimum nearest neighbor classification. Regarding the training time, learning the ITML metric was the fastest and learning NCA and LMNN were the slowest ones. Due to computational constrains we set the parameters of LMNN metric to reduce the size of features to 100. NCA metric reduces the dimension of features to the number of categories for each tasks: 27 for style classification, 23 for artist classification and 10 for genre classification. We randomly picked 3000 samples, which we used for metric learning. These samples follow the same distribution as original data and are not used for classification experiments.

Classification Experiments

For the purpose of metric learning, we conducted experiments with labels for three different tasks of style, genre and artist prediction. In following sections we investigate the performance of these metrics on different features for classification of aforementioned concepts.

We learned all the metrics in sectionA.1.3 for all 27 styles of paintings in our dataset (e.g. Expressionism, Realism, etc.). However, we did not use all the genres for learning metrics. In fact in our dataset we have 45 genres, some of which have less than 20 images. This makes the metric learning impractical and highly biased toward genres with larger number of paintings. Because of this issue, we focus on 10 genres with more than 1500 paintings. These genres are listed in table A.1. In all experiments we conducted 3 fold cross validation and reported the average accuracy over all partitions. We found the best value for penalty term in SVM (which is equal to 10) by three fold cross validation. In the next three sections, we explain settings and findings for each task independently.

Style Classification

Table A.2 contains the result (accuracy percentage) of style classification (SVM) after applying different metrics on a set of features. Columns correspond to different features and rows are

[6]http://www.ics.uci.edu/ fowlkes/

Metric / Features	GIST	Classemes	Picodes	CNN	Dim.
Baseline	28.10	49.98	49.63	35.14	512
Boost	31.01	**57.87**	**57.35**	46.14	512
ITML	33.10	57.86	57.28	46.80	512
LMNN	**39.06**	54.96	54.42	**49.98**	100
MLKR	32.81	54.29	42.79	45.02	512
NCA	30.39	51.38	52.74	49.26	10

Table A.3: Accuracy for the task of genre classification

different metrics that are used for projecting features before learning style classifiers. In order to quantify the improvement by learning similarity metrics, we conducted a baseline experiment (first row in the table) as the following: For each type of features, we learn a set of one-vs-all classifiers on raw feature vectors. Generally Boost metric learning and ITML approaches give the highest in accuracy for the task of style classification over different visual features. However the greatest improvement over the baseline is gained by application of Boost metric on Classeme features. We visualized the confusion matrix for the task of style classification, when we learn Boost metric on Classeme features.

Figure A.5 shows this matrix, where red represents higher values. Further analysis of some confusions that are captured in this matrix result in interesting findings. In the rest of this paragraph we explain some of these cases. First, we found that there is a big confusion between "Abstract expressionism" (first row) and "Action paintings" (second column). Art historians verify the fact that this confusion is meaningful and somehow expected. "Action painting" is a type or subgenre of "abstract expressionism" and are characterized by paintings created through a much more active process– drips, flung paint, stepping on the canvas.

Another confusion happens between "Expressionism" (column 10) and "Fauvism" (row 11), which is actually expected based on art history literature. "Mannerism" (row 14) is a style of art during the (late)"Renaissance" (column 12), where they show unusual effect in scale and are less naturalistic than "Early Renaissance". This similarity between "Mannerism" (row 14) and "Renaissance" (column 12) is captured by our system as well where results in confusion during style classification. "Minimalism" (column 15) and "Color field paintings"(6th row) are mostly confused with each other. We can agree on this finding as we look at members of

these styles and figure out the similarity in terms of simple form and distribution of colors. Lastly some of the confusions are completely acceptable based on the origins of these styles (art movements) that are noted in art history literature. For example, “Renaissance”(column 18) and “Early Renaissance”(row 9); “Post Impressionism” (column 21) and “Impressionism”(row 13); “Cubism” (8th row) and “Synthetic Cubism” (column 26). Synthetic cubism is the later act of cubism with more color continued usage of collage and pasted papers, but less linear perspective than cubism.

Genre Classification

We narrowed down the list of all genres in our dataset (45 in total) to get a reasonable number of samples for each genre (10 selected genres are listed in table A.1). We trained ten one-vs-all SVM classifiers and compare their performance in Table A.3. In this table columns represent different features and rows are different metric that we used to compute the distance. As table A.3 shows we achieved the best performance for genre classification by learning Boost metric on top of Classeme features. Generally the performance of these classifiers are better than classifiers that we trained for style classification. This is expected as the number of genres is less than the number of styles in our dataset.

Figure A.6 shows the confusion matrix for classification of genre by learning Boost metric, when we used Classeme features. Investigating the confusions that we find in this matrix, reveals interesting results. For example, our system confuses “Landscape” (5th row) with “Cityspace” (2nd column) and “Genre paintings” (3rd column). However, this confusion is expected as art historians can find common elements in these genres. On one hand “Landscape” paintings usually show rivers, mountains and valleys and there is no significant figure in them; frequently very similar to “Genre paintings” as they capture daily life. The difference appears in the fact that despite the “Genre paintings”, “Landscape” paintings are idealized. On the other hand, “Landscape” and “Cityspace” paintings are very similar as both have open space and use realistic color tonalities.

Metric / Features	GIST	Classemes	Picodes	CNN	Dim.
Baseline	17.58	45.29	45.82	20.38	512
Boost	**25.65**	**57.76**	**55.50**	29.65	512
ITML	19.95	51.79	53.93	**31.04**	512
LMNN	20.41	53.99	53.92	30.92	100
MLKR	21.22	49.61	19.54	21.77	512
NCA	18.80	53.70	53.81	22.26	23

Table A.4: Accuracy for the task of artist classification

Artist Classification

For the task of the artist classification, we trained one-vs-all SVM classifiers for each of 23 artists. For each test image, we determine its artist by finding the classifier that produces the maximum confidence. Table A.4 shows the performance of different combinations of features and metrics for this task. In general learning Boost metric improves artist classification better than all other metrics, except the case of CNN features where learning ITML metric gained the best performance. We plotted the confusion matrix of this classification task in figure A.7. In this plot, some confusions between artists are clearly reasonable. We investigated two cases:

First case, “Claude Monet”(5th row) and “Camille Pissaro”(3rd column). Both of these Impressionist artists who lived in the late nineteen and early twentieth centuries. Interestingly, based on art history literature Monet and Pissaro became friends when they both attended the ”Académie Suisse” in Paris. This friendship lasted for a long time and resulted in some noticeable interactions between them. Second case, paintings of “Childe Hassam”(4th row) are mostly confused with ones from “Monet”(5th column). This confusion is acceptable as Hassam is an American Impressionist, who declared himself as being influenced by French Impressionists. Hassam called himself an “Extreme Impressionist”, who painted some flag-themed artworks similar to Monet.

By looking at reported performances in tables A.2- A.4, we conclude that, all three classification tasks can benefit from learning the appropriate metric. This means that we can improve the accuracy of baseline classification by learning metrics independent of the type of visual feature or the concept that we are classifying painting based on. Experimental results show that, independent of the task, NCA and MLKR approaches are performing worse than other metrics. Additionally, Boost metric always gives the best or the second best results for all classification

Task / Features	GIST	Classemes	Picodes	CNN
Style	20.21	**37.33**	**33.27**	21.99
Genre	35.94	**58.29**	**56.09**	47.05
Artist	30.37	**59.37**	**55.65**	33.62

Table A.5: Classification performance for metric fusion methodology

tasks.

Regarding analysis of importance of features, we can verify that Classeme and Picode features are better image representations for classification purposes. Based on these classification experiments, we claim that Classemes and Picodes features perform better than CNN features. This is rooted in the fact that amount of supervision for training Classeme and Picodes is more than CNN training. Also, unlike Classeme and Picodes, CNN feature is designed to categorize the object insides a given bounding box. However, in the case of paintings we cannot assume that all the bounding boxes around the objects are given.

Integration of Features and Metrics

We investigated the performance of different metric learning approaches and visual features individually. In the next step, we find out the best performance for aforementioned classification tasks by combining different visual features. Toward this goal, we followed two strategies. First, for a given metric, we project visual features by applying the metric and concatenate these projected visual features together. Second, we fixed the type of visual feature that we use and project it with the application of different metrics and concatenate these projections all together. Having this larger feature vectors (either of two strategies), we train SVM classifiers for three tasks of Style, Genre and Artist classification. Table A.6 shows the results of these experiments where we followed the earlier strategy and table A.5 shows the results of the later case. In general we get better results by fixing the metric and concatenating the projected feature vectors (first strategy).

The work of Bar et al [8] is the most similar to ours and we compare our final results of these experiments with their reported performance. [8] only performed the task of style classification on half of the images in our dataset and achieved the accuracy of 43% by using two variations of

Concept / Metric	Boost	ITML	LMNN	MKLR	NCA
Style	41.74	**45.05**	**45.97**	38.91	40.61
Genre	**58.51**	**60.28**	58.48	55.79	54.82
Artist	**61.24**	60.46	**63.06**	53.19	55.83

Table A.6: Classification results for feature fusion methodology.

PiCoDes features and two layers of CNN. However we outperform their approach by achieving 45.97 % accuracy for the task of style classification when we used LMNN metric to project GIST, Classeme, PiCoDes and CNN features and concatenate them all together as it is reported in the third column of table A.6.

Our contribution goes beyond outperforming state-of-the-art by learning a more compact feature representation. In this work, our best performance for style classification happens when we concatenate four 100-dimensional feature vectors. This results in a 400 dimensional feature vectors that we train SVM classifiers on top of them. However [8] extract a 3882 dimensional feature vector to their best reported performance. As a result we not only outperform the state-of-the-art, but presented a better image representation that reduces the amount of space by 90%. Our efficient feature vector is an extremely useful image representation that gains the best classification accuracy and we consider its application for the task of image retrieval as future work.

A.1.8 Conclusion and Future Works

In this chapter we investigated the applicability of metric learning approaches and performance of different visual features for learning similarity in a collection of fine-art paintings. We implemented meaningful metrics for measuring similarity between paintings. These metrics are learned in a supervised manner to put paintings from one concept close to each other and far from others. In this chapter we used three concepts: Style, Genre and Artist. We used these learned metrics to transform raw visual features into another space that we can significantly improve the performance for three important tasks of *Style, Genre* and *Artist classification*. We conducted our comparative experiments on the largest publicly available dataset of fine-art paintings to evaluate the performance for the aforementioned tasks.

We conclude that:

- Classeme features show the superior performance for all three tasks of Style, Genre or Artist classification. This superior performance is independent of the type of metric that has been learned.

- In the case of working on individual type of visual features, Boost metric and Information Theoretic Metric Learning(ITML) approaches improve the accuracy of classification tasks across all features.

- For the case of using different types of features all together(feature fusion), Large-Margin Nearest-Neighbor(LMNN) metric learning achieves the best performance for all classification experiments.

- By learning LMNN metric on Classeme features, we find an optimized representation that not only outperforms state-of-the art for the task of style classification, but reduce the size of feature vector by 90%.

We consider verification of applicability of this representation for the task of image retrieval and recommendation systems as future work. As other future works we would like to learn metrics based on other annotation(e.g. time period).

A.2 Toward Automated Discovery of Artistic Influence

1. Babak Saleh, Kanako Abe, Ahmed Elgammal: Knowledge Discovery of Artistic Influences: A Metric Learning Approach, International Conference on Computational Creativity (ICCC) 2014.

2. Babak Saleh, Kanako Abe, Ravneet Singh Arora, Ahmed Elgammal: Toward Automated Discovery of Artistic Influence, Multimedia Tools and Applications, Springer, August 2014.

Considering the huge amount of art pieces that exist, there is valuable information to be discovered. Examining a painting, an expert can determine its style, genre, and the time period that the painting belongs. One important task for art historians is to find influences and connections between artists. Is influence a task that a computer can measure? The contribution of this paper is in exploring the problem of computer-automated suggestion of influences between artists, a problem that was not addressed before in a general setting. We first present a comparative study of different classification methodologies for the task of fine-art style classification. A two-level comparative study is performed for this classification problem. The first level reviews the performance of discriminative vs. generative models, while the second level touches the features aspect of the paintings and compares semantic-level features vs. low-level and intermediate-level features present in the painting. Then, we investigate the question "Who influenced this artist?" by looking at his masterpieces and comparing them to others. We pose this interesting question as a knowledge discovery problem. For this purpose, we investigated several painting-similarity and artist-similarity measures. As a result, we provide a visualization of artists (Map of Artists) based on the similarity between their works

A.2.1 Introduction

How do artists describe their paintings? They talk about their works using several different concepts. The elements of art are the basic ways in which artists talk about their works. Some of the *elements of art* include space, texture, form, shape, color, tone and line [60]. Each work of art can, in the most general sense, be described using these seven concepts. Another important descriptive set is the *principles of art*. These include movement, unity, harmony, variety,

Figure A.8: An example of an often cited comparison in the context of influence. Left: Diego Velázquez's Portrait of Pope Innocent X (1650), and, Right: Francis Bacon's Study After Velázquez's Portrait of Pope Innocent X (1953). Similar composition, pose, and subject matter but a different view of the work.

balance, contrast, proportion, and pattern [60]. Other topics may include subject matter, brushstrokes, meaning, and historical context. As seen, there are many descriptive attributes in which works of art can be talked about.

One important task for art historians is to find influences and connections between artists. By doing so, the conversation of art continues and new intuitions about art can be made. An artist might be inspired by one painting, a body of work, or even an entire style of art. Which paintings influence each other? Which artists influence each other? Art historians are able to find which artists influence each other by examining the same descriptive attributes of art which were mentioned above. Similarities are noted and inferences are suggested.

It must be mentioned that determining influence is always a subjective decision. We will not know if an artist was ever truly inspired by a work unless he or she has said so. However, for the sake of finding connections and progressing through movements of art, a general consensus is agreed upon if the argument is convincing enough. For example, Figure A.8 illustrates a commonly cited comparison for studying influence, in the work of Francis Bacon's Study After Velázquez's Portrait of Pope Innocent X (1953), where similarity is clear in composition, pose,

and subject matter.

Is influence a task that a computer can measure? In the last decade there have been impressive advances in developing computer vision algorithms for different object recognition-related problems including: instance recognition, categorization, scene recognition, pose estimation, etc. When we look into an image we not only recognize object categories, and scene category, we can also infer various aesthetic, cultural and historical aspects. For example, when we look at a fine-art paining, an expert, or even an average person can infer information about the style of that paining (e.g. Baroque vs. Impressionism), the genre of the painting (e.g. a portrait or a landscape), or even can guess the artist who painted it. People can look at two painting and find similarities between them in different aspects (composition, color, texture, subject matter, etc.) This is an impressive ability of human perception for learning and judging complex aesthetic-related visual concepts, which for long have been thought not to be a logical process. In contrast, we tackle this problem using a computational methodology approach, to show that machines can in fact learn such aesthetic concepts.

Although there has been some research on automated classification of paintings e.g. [5, 26, 28, 99, 69], however, there is almost no research done on computer-based measuring and determining of influence between artists. Measuring influence is a very difficult task because of the broad criteria for what influence between artists can mean. As mentioned earlier, there are many different ways in which paintings can be described. Some of these descriptions can be translated to a computer. For example, Li et al [99] proposed automated way for analyzing brushstrokes to distinguish between Van Gogh and his contemporaries.

For the purpose of this paper, we do not focus on a specific element of art or principle of art but instead we focus on finding and suggesting new comparisons by experimenting with different similarity measures and features.

What is the benefit of the study of automated methods of analyzing painting similarity and artistic influences? By including a computer quantified judgement about which artists and paintings may have similarities, it not only finds new knowledge about which paintings are connected using a mathematical criteria, but also keeps the conversation going for artists. It challenges people to consider possible connections in the timeline of art history that may have never been seen before. *We are not asserting truths but instead suggesting a possible path*

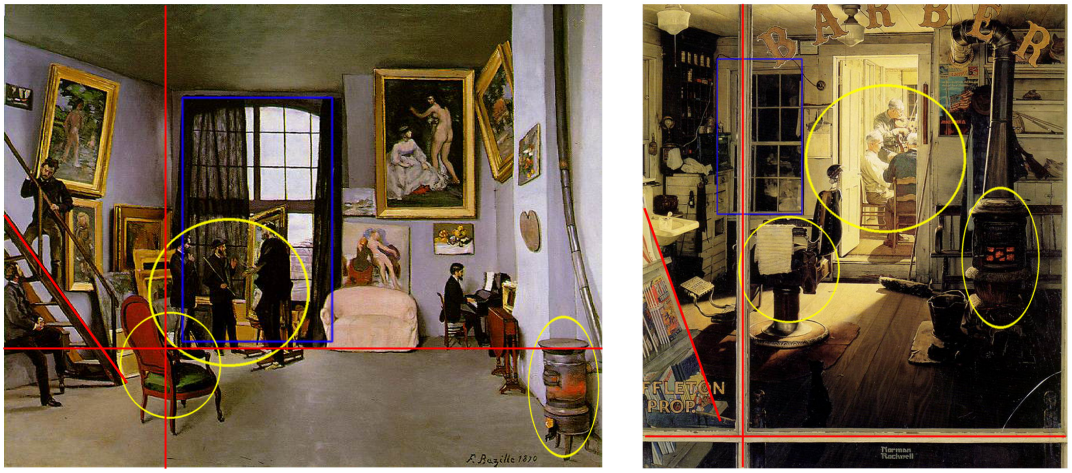

Figure A.9: Frédéric Bazille's *Studio 9 Rue de la Condamine* (left) and Norman Rockwell's *Shuffleton's Barber Shop* (right). The composition of both paintings is divided in a similar way. Yellow circles indicate similar objects, red lines indicate composition, and the blue square represents similar structural element. The objects seen – a fire stove, three men clustered, chairs, and window are seen in both paintings along with a similar position in the paintings. After browsing through many publications and websites, we conclude that this comparison has not been made by an art historian before.

towards a difficult task of measuring influence.

Besides the scientific merit of the problem, there are various application-oriented motivations. With the increasing volumes of digitized art databases on the internet comes the daunting task of organization and retrieval of paintings. There are millions of paintings present on the internet. To manage properly the databases of these paintings, it becomes very essential to classify paintings into different categories and sub-categories. This classification structure can be utilized as an index and thus can improve the speed of retrieval process. Also it will be of great significance if we can infer new information about an unknown painting using already existing databases of paintings, and as a broader view can infer high-level information like influences between painters.

Although the meaning of a painting is unique to each artist and is completely subjective, it can somewhat be measured by the symbols and objects in the painting. Symbols are visual words that often express something about the meaning of a work as well. For example, the works of Renaissance artists such as Giovanni Bellini and Jan Van-Eyck use religious symbols such as a cross, wings, and animals to tell stories in the Bible. This shows the need for an

object-based representation of images. We should be able to describe the painting from a list of many different object classes. By having an object-based representation, the image is described in a high-level semantic as opposed to low-level features such as color and texture, which facilitates suggesting influences based on subject matter. Paintings do not necessarily have to look alike, but if they do, or have reoccurring objects (high-level semantics), then they might be considered similar. If influence is found by looking at similar characteristics of paintings, the importance of finding a good similarity measure becomes prominent. Time is also an essential factor in determining influence. An artist cannot influence another artist in the past. Therefore the linearity of paintings cuts down the possibilities of influence.

The contribution of this chapter is in exploring the problem of computer-automated suggestion of influences between artists, a problem that was not addressed before in a general setting. From a machine-learning point of view, we approach the problem as an unsupervised knowledge discovery problem. Our methodology is based on three components: 1) studying different representations of painting to determine which is more useful for the task of influence detection; 2) measuring similarity between paintings; 3) studying different measures of similarity between artists. We collected a comprehensive painting dataset for conducting our study. The data set contains 1710 high-resolution images of paintings by 66 artist spanning the time period of 1412-1996 and containing 13 painting styles. We also collect a ground-truth data set for the task of artistic influences, which mainly contains positive influences claimed by art historian. This ground-truth is only used for the overall evaluation of our discovered/suggested influences, and is not used in the learning or knowledge-discovery.

We hypothesis that a high-level semantic representation of painting would be more useful for the task of influence detection. However, evaluating such a hypothesis requires comparing the performance of different features and representation in detecting influences against a ground-truth of artistic influences, containing both positive and negative example. However, because of the limited size of the available ground-truth data, and the lack of negative examples in it, it is not useful for comparing different features and representations. Instead we resort to a highly correlated task, which is classifying painting style. The hypothesis is that features and representations that are good for style classification (which is a supervised learning problem),

would also be good for determining influences (which is an unsupervised problem). Therefore, we performed a comprehensive comparative study of different features and classification models for the task of classifying painting style among seven different styles. This study is described in details in Sec A.2.4. The conclusion of this study confirms our hypothesis that high-level semantic features would be more useful for the task of style classification, and hence useful for determining influences.

Using the right features to represent the painting paves the way to judge similarity between paintings in a quantifiable way. Figure A.9 illustrates an example of similar paintings detected by our automated methodology; Frédéric Bazille's *Studio 9 Rue de la Condamine* (1870) and Norman Rockwell's *Shuffleton's Barber Shop* (1950). After browsing through many publications and websites, we concluded, to the best of our knowledge, that this comparison has not been made by an art historian before. The painting might not look similar at the first glance, however, a closer look reveals striking similarity in composition and subject matter, that is detected by our automated methodology (see caption for details). Other example similarity can be seen in Figures A.14 & A.15.

Measuring similarity between painting is fundamental to discover influences, however, it is not clear how painting similarity might be used to suggest influences between artist. The paintings of a given artist can span extended period of time and can be influenced by several other contemporary and prior artists. Therefore, we investigated several artist distance measures to judge similarity in their work and suggest influences. As a result of this distance measures, we can achieve visualizations of how artists are similar to each other, which we denote by a map of artists.

The paper is structured as follows: Section A.2.2 provides a literature survey on the topic of computer-based methods for analyzing painting. Section A.2.3 describes the data set used in our study. Section A.2.4 describes our comparative study for the task of painting style classification, including the methodologies, features and the results. Section A.2.9 describes our methodology for judging artistic influence. Section A.2.10 represents qualitative and quantitative evaluation of our automated influence study.

A.2.2 Related Works

There is little work done in the area of automated fine-art classification. Most of the work done in the problem of paintings classification utilizes low-level features such as color, shades, texture and edges. Lombardi [100] presented a comprehensive study of the performance of such features for paintings classification. In that paper the style of the painting was identified as a result of recognizing the painter. Sablatnig et al. [128] used brushstroke patterns to define structural signature to identify the artist style. Khan et al. [51] used a Bag of Words (BoW) approach with low-level features of color and shades to identify the painter among eight different artists. In [135] and [78] similar experiments with low-level features were conducted. Unlike most of the previous works that focused on inferring the artist from the painting, our goal is to directly recognize the style of the painting, and discover artist similarity and influences, which are more challenging tasks.

Carneiro et al. [29] recently published the dataset "PRINTART" on paintings along with primarily experiments on image retrieval and painting style classification. They provided three levels of annotation for the "PRINTART" dataset: Global, Local and Pose annotation. However this dataset contains only monochromatic artistic images. We present a new dataset which has chromatic images and its size is about double the "PRINTART" dataset covering a more diverse set of styles and topics. Carneiro et al. [29] showed that the low-level texture and color features exploited for photographic image analysis are not as effective because of inconsistent color and texture patterns describing the visual classes (e.g. humans) in artistic images.

Carneiro et al. [29] define artistic image understanding as a process that receives an artistic image and outputs a set of global, local and pose annotations. The global annotations consist of a set of artistic keywords describing the contents of the image. Local annotations comprise a set of bounding boxes that localize certain visual classes, and pose annotations consisting of a set of body parts that indicate the pose of humans and animals in the image. Another process involved in the artistic image understanding is the retrieval of images given a query containing an artistic keyword. In. [29] an improved inverted label propagation method was proposed that produced the best results, both in the automatic (global, local and pose) annotation and retrieval problems.

Carneiro et. al. [28] targeted the problem of annotating an unseen image with a set of global labels, learned on top of annotated paintings. Furthermore, for a given set of visual classes, they are able to retrieve the painting which shows the same characteristics. They have proposed a graph-based learning algorithm based on the assumption that visually similar paintings share same annotation. They formulated the global annotation problem with a combinatorial harmonic approach, which computes the probability that a random walk starting at the test image first reaches each of the database samples. However all the samples are from fifteen to seventeen century and focused on religious themes.

Graham et. al. [69] posed the question of finding the way we perceive two artwork as similar to each other. Toward this goal, they acquired strong supervision of human experts to label similar paintings. They apply multidimensional scaling methods to paired similar paintings from either Landscape or portrait/still life and showed that similarity between paintings can be interpreted as basic image statistics. In the experiments they show that for landscape paintings, basic grey image statistics is the most important factor for two artwork to be similar. For the case of still life/portrait most important elements of similarity are semantic variables, for example representation of people.

Unlike the case of ordinary images, where color and texture are proper low-level features to be used for a diverse set of tasks (e.g. classification), these might not describe paintings well. Color and texture features are highly prone to variations during digitization of paintings. In the case of color, it also lacks fidelity due to aging. The effect of digitization on the computational analysis of paintings is investigated in great depth by Polatkan et. al [127].

The aforementioned reasons make the brushstrokes more meaningful features for describing paintings. Li et al. [99] used fully automatic extracted brushstrokes to describe digitized paintings. Their novel feature extraction method is developed by the integration of edge detection and clustering-based segmentation. Using these features they found that regularly shaped brushstrokes are tightly arranged, creating a repetitive and patterned impression that can represent Van Gogh style and help to distinguish his work from his contemporaries. They have conducted a set of analysis based on 45 digitized oil paintings of Van Gogh from museum's collections. Due to small number of samples, and to avoid overfitting, they state this problem as a hypothesis testing rather than classification. They hypothesize which factors are eminent

in Van Gogh style comparing to his contemporaries and tested them by statistical approaches on top of brushstroke features.

Cabral et al [26] approached the problem of ordering paintings and estimating their time period. They formulated this problem as embedding paintings into a one dimensional manifold and tried two different methods: on one hand, they applied unsupervised embedding using Laplacian Eignemaps [11]. To do so they only need visual features and defined a convex optimization to map paintings to a manifold. This approach is very fast and do not need human expertise, but its accuracy is low. On the other hand, since some partial ordering on paintings is available by experts, they use these information as a constraint and used Maximum Variance Unfolding [172] to find a proper space, capturing more accurate ordering of paintings.

A.2.3 Dataset

Our dataset contains a total of 1710 images of art works by 66 artists, chosen from Mark Harden's Artchive database of fine-art [73]. Each image is annotated with the artist's first name, last name, title of work, year made, and style. The majority of the images are of the full work while a few are details of the work. We are primarily dealing with paintings but we have included very few images of sculptures as well. The artist with the largest number of images is Paul Cézanne with 140 images, and the artist with the least number of works is Hans Hoffmann with 1 image.

The artists themselves ranged from 13 different styles throughout art history. These include, with no specific order, Expressionism (10 artists), Impressionism (10), Renaissance (12), Romanticism (5), Cubism (4), Baroque (5), Pop (4), Abstract Contemporary (7), Surrealism (2), American Modernism (2), Post-Impressionism (3), Symbolism (1), and Neoclassical (1). The number in the parenthesis refers to the number of artists in each style category. Some styles were condensed such as *Abstract Contemporary*, which includes works in the *Abstract Expressionism, Contemporary,* and *De Stijl* periods. The *Renaissance* period has the most images (336 images) while *American Modernism* has the least (23 images). The average number of images per style is 132. The earliest work is a piece by Donatello in 1412, while the most recent work is a self portrait by Gerhard Richter done in 1996. The earliest style is the *Renaissance* period with artists like Titian and Michelangelo during the 14th to 17th century. As for the most recent

Figure A.10: Examples of paintings from thirteen styles: Renaissance, Baroque, Neoclassical, Romanticism, Impressionism, Post-Impressionism, Expressionism, Cubism, Surrealism, Symbolism, American Modernism, Pop, and Abstract Contemporary.

style, art movements tend to overlap more in recent years. Richter's painting from 1996 is in the *Abstract Contemporary* style.

A.2.4 Painting-Style Classification: A Comparative Study

In this section we present the details of our study on painting style classification. The problem of painting style classification can be stated as: Given a set of paintings for each painting style, predict the style of an unknown painting. A lot of work has been done so far on the problem of image category recognition, however the problem of painting classification proves quite different than that of image category classification. Paintings are differentiated, not only by contents, but also by style applied by a particular painter or school of painting or by the age when they were painted. This makes painting classification problem much more challenging than the ordinary image category recognition problem.

In this study we will approach the problem of painting style classification from a supervised learning perspective. A two-level comparative study is conducted for this classification problem. The first level reviews the performance of discriminative vs. generative models, while the second level touches the feature aspects of the paintings and compares semantic-level features vs. low-level and intermediate-level features present in the painting.

For experimental purposes seven fine-art styles are used, namely *Renaissance, Baroque,*

Impressionism, Cubism, Abstract, Expressionism, and Popart. Various different sets of comparative experiments were performed focused on evaluation of classification accuracy for each methodology. We evaluated three different methodologies, namely:

1. Discriminative model using a Bag-of-Words (BoW) approach
2. Generative model using BoW
3. Discriminative model using Semantic-level features

As shown in Figure A.2.4, these three models differ in terms of the classification methodology, as well as the type of features used to represent the painting. The Discriminative Semantic-Level model applies a discriminative machine learning model upon features capturing semantic information present in a painting, while Discriminative and Generative BoW models employs discriminative and generative machine learning models, respectively, on the Intermediate level features represented using a BoW model.

A generative model has the property that it specifies a joint probability distribution over observed samples and their labels. In other words, a generative classifier learns a model of joint probability distribution $p(x, y)$, where x denotes the observed samples and y are the labels. Bayes rule can be applied to predict the label y for a given new sample x, which is determined by the probability distribution $p(y|x)$. Since a generative model calculates the distribution $p(x|y)$ as an intermediate step, these can be used to generate random instances x conditioned on target labels y. A discriminative model, in contrast, tries to estimate the distribution $p(y|x)$ directly from the training data. Thus, a discriminative model bypasses the calculation of joint probability distribution $p(x, y)$ and avoids the use of Bayes rule. We refer the reader to [112] for a comprehensive comparison of both learning models.

It is also very important to make distinction between Low, Intermediate, and Semantic -level features at this stage. Low-level features capture directly the formal characteristics of paintings such as color, texture, edges, light etc. The average intensity of all the pixels, color histogram representing color composition of paintings and number of edges are examples of low-level features that capture the formal elements light, color and edges respectively. Intermediate-level

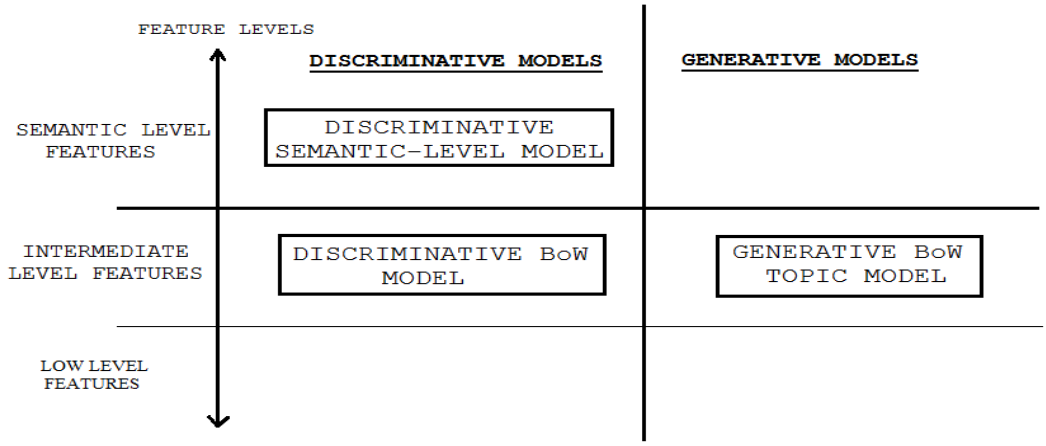

Figure A.11: Illustrative diagram of approaches for style classification of paintings

features apply local-level descriptors like SIFT [166] and CSIFT on various regions of an image. Local level descriptors instead of summarizing the whole image, represents localized regions of an image. A Bag of Words model is applied to generate an intermediate representation of the image. A Bag of Words model first creates fixed number of clusters from the localized regions of all the images (a codebook of visual vocabulary) and further represents each image by the histogram capturing the frequency of the code words in that image. Semantic-level features capture the semantic content classes such as water, sand, cars etc. present in an image. Thus, such frequency of semantic classes can help us in ranking images according to their semantic similarity. A feature vector where each element denotes the probability of existence of a semantic class is an example of semantic feature. It is worth noting that, instead of using low-level features like color, light, shades and texture our study is focused on intermediate-level features (BoW features) and semantic-level features.

We hypothesize the following claims 1) Semantic-level information contained in a painting can be very well utilized for the task of classification and 2) Generative models like Topic models are very much capable of capturing the thematic structure of a painting. It is easy to visualize a topic or theme in the case of documents. For documents, a topic can be a collection of particular set of words. For example, a science topic is characterized by the collection of words like atom, electrons, protons etc. For images represented by a Bag of Word model, each

word is represented by the local level descriptor used to describe the image. Thus a collection of particular set of such similar regions can constitute a topic. For example, collection of regions representing mainly straight edges can constitute the topic trees. Similarly, set of regions having high concentration of blue color can form up a theme related to sky or water.

The following subsections describe the details of the compared methodologies.

A.2.5 Discriminative Bag-of-Words model

Bag of Words(BoW) [153] is a very popular model in text categorization to represent documents, where the order of the words does not matter. BoW was successfully adapted for object categorization, e.g. in [64, 160, 176]. Typical application of BoW on an image involves several steps, which includes:

1) Locating interest points in an image

2) Representation of such points/regions using feature descriptors

3) Codebook formation using K-Means clustering, to obtain a "dictionary" or a codebook of visual words.

4) Vector quantization of the feature descriptor; each descriptor is encoded by its nearest visual word from the codebook.

5) Generate an intermediate-level representations for each image using the codebook, in the form of a histogram of the visual words present in each image.

6) Train a discriminative classifier on the intermediate training feature vectors for each class.

7) For classification, the trained classifier is applied on the BoW feature vector of a test image.

Thus, the end result of a Bag of Words model is a histogram of words, which is used as an intermediate-level feature to represent a painting. In our study, we applied a Support Vector Machine (SVM) classifier [32] on a code-book trained on images from our dataset. We used two variant of the widely used Scale Invariant Feature Transform "SIFT" features [166] called Color SIFT (CSIFT) [166] and opponent SIFT (OSIFT) [166] as local features. The SIFT [166] is invariant to image scale, rotation, affine distortion and illumination. It uses edge

orientations to define a local region and also utilizes the gradient of an image. Also, the SIFT descriptor is normalized and hence is also immune to gradient magnitude changes. CSIFT and opponent SIFT (OSIFT) extends SIFT features for color images, which is essential for the task of painting-style classification. In an earlier study by Van De Sande et al [166] opponent SIFT was shown to outperform other color SIFT variants in image categorization tasks.

A.2.6 Discriminative Semantic-level model

In this approach a discriminative model is employed on top of semantic-level features. Seeking semantic-level features, we extracted the Classeme feature vector [165] as the visual feature for each painting. Classeme features are output of a set of classifiers corresponding to a set of C category labels, which are drawn from an appropriate term list, defined in [165], and not related to our fine-art context. For each category $c \in \{1 \cdots C\}$, a set of training images was gathered by issuing a query on the category label to an image search engine. After a set of coarse feature descriptors (Pyramid HOG, GIST) is extracted, a subset of feature dimensions was selected [165]. Using this reduced dimension features, a one-versus-all classifier ϕ_c is trained for each category. The classifier output is real-valued, and is such that $\phi_c(x) > \phi_c(y)$ implies that x is more similar to class c than y is. Given an image x, the feature vector (descriptor) used to represent it is the Classeme vector $[\phi_1(x), \cdots, \phi_C(x)]$. The Classeme feature is of dimensionality $N = 2569$.

We used such feature vectors to train a Support Vector Machine (SVM) [32] classifier for each painting genre. We hypothesize that Classeme features are suitable for representing and summarizing the overall contents of a painting since it captures semantic-level information about object presence in a painting encoded implicitly in the output of the pre-trained classifiers.

A.2.7 Generative Bag-of-Words Topic model

Generative topic model uses Latent Dirichlet Allocation (LDA) [35]. In studies [56] and [83], LDA and Probabilistic Latent Semantic Analysis (pLSA) topic models have been applied for object categorization, localization and scene categorization. This paper is the first evaluation of such models in the domain of fine-art categorization.

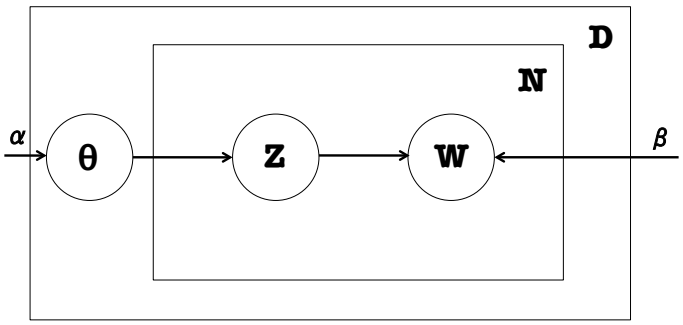

Figure A.12: Graphical model representing Latent Dirichlet Allocation

For the purpose of our study, we used Latent Dirichlet Allocation (LDA [35]) topic model and applied it on BoW representation of paintings using both CSIFT and OSIFT features. In LDA, each item is represented by a finite mixture over a set of topics and each topic is characterized by a distribution over words. Figure A.2.7 shows a graphical model for the image generation process. As shown in the model, parameter Θ defines the topic distribution for each image (total number of images is D.) Θ is determined by Dirichlet parameter α, β and represents the word distribution for each topic. The total number of words is N. To use LDA for the classification task, we build model for each of the styles in our framework. First step is to represent each training image by a quantized vector using Bag-of-Words model described earlier. This vector quantized representation of each image is used for parameter estimation using Variational Inference. Thus, we will get LDA parameters Θ_c and β_c for each category c. Once we have a new test image, d, we can infer the parameter Θ_{cd} for each category and $p(d|\Theta_{cd}, \beta_c)$ is used as the likelihood of the image belonging to a particular class c.

A.2.8 Style Classification Results

For the task of Style classification of paintings, we focus on a subset of our dataset that contains seven categories of paintings namely Abstract, Baroque, Renaissance, Pop-art, Expressionism, Impressionism and Cubism. Each category consists of 70 paintings. For each of the following experiments five-fold cross-validation was performed, with 20% of the images chosen for testing purpose in each fold.

For codebook formation, Harris-Laplace detector [151] is used to find the interest points. For efficient computation the number of interest points for each painting is restricted to 3000.

Confusion(%)	Baroque	Abstract	Renaissance	Pop-Art	Expressionism	Impressionism	Cubism
Baroque	87.5	0	14.3	0	5.3	17.8	1.78
Abstract	0	64	0	7.1	7.1	1.8	1.9
Renaissance	5.4	0	64.3	5.35	14.3	3.5	1.8
Pop-Art	0	1.78	1.8	73.1	0	3.5	1.8
Expressionism	1.8	20.2	7.1	3.6	48.2	17.8	12.9
Impressionism	5.36	8	9	5.3	17.8	48.2	9.2
Cubism	0	6	3.5	5.3	7.1	7.1	72.4

Table A.7: Confusion matrix for Discriminative Semantic Model

Confusion(%)	Baroque	Abstract	Renaissance	Pop-Art	Expressionism	Impressionism	Cubism
Baroque	71.4	0	12.9	0	8.5	17.1	0
Abstract	0	48	5.8	10	8.5	5.7	7.1
Renaissance	18.6	6.7	41.4	0	5.8	9.3	18.5
Pop-Art	0	15	0	70	11.5	9.3	15.7
Expressionism	0	15	18.6	2.8	28.5	12.9	13
Impressionism	8.5	8.6	3.7	8.6	17.2	45.7	11.4
Cubism	1.5	6.7	17.6	8.6	20	0	34.3

Table A.8: Discriminative BoW using CSIFT

Standard K-means Clustering algorithm is used to build a Codebook of size 600 words. SVM classifier is trained on both intermediate-level and semantic-level descriptors. For SVM, we use Radial Basis function (RBF) kernels. To determine parameters for the SVM, the grid search algorithm implemented by [32] is employed. Grid search algorithm uses cross-validation to pick up the optimum parameter values. Also this process is preceded by scaling of dataset descriptors. For experiments with LDA, David Beli's C-code [35] is used for the task of parameter estimation and inference. This C-code uses Variational Inference technique, which tries to estimate parameters β and Θ using a similar and simpler model. For parameter estimation alpha is set to be 0.1 and LDA code is set to estimate the value of α during the estimation process.

We evaluated and tested the three models on our dataset, and calculated and compared the classification accuracy for each of them. Table A.7 shows the confusion matrix of the Discriminative Semantic Model over the five-fold cross validation. The overall accuracy achieved is 65.4 %. Table A.8 and A.9 show the confusion matrices for the discriminative BoW model with CSIFT and OSIFT features respectively. Overall accuracy achieved is 48.47% and 56.7% respectively. Table A.10 and A.11 show the confusion matrices for the generative topic model

Confusion(%)	Baroque	Abstract	Renaissance	Pop-Art	Expressionism	Impressionism	Cubism
Baroque	82.1	0	10.7	0	14.3	17.9	3.6
Abstract	0	54.2	3.6	7.1	7.1	3.6	7.1
Renaissance	3.6	0	64.3	3.6	21	0	7.1
Pop-Art	0	12.5	3.6	75	0	0	17.9
Expressionism	0	16.7	0	3.6	36	10.7	28.6
Impressionism	14.3	8.33	7.2	3.6	10.7	57.1	7.1
Cubism	0	4.2	10.8	7.1	14.3	10.7	28.6

Table A.9: Discriminative BoW using OSIFT

Confusion(%)	Baroque	Abstract	Renaissance	Pop-Art	Expressionism	Impressionism	Cubism
Baroque	86.6	0	14.3	0	14.3	7.1	7.1
Abstract	0	58.3	7.1	26.6	0	7.1	14.3
Renaissance	6.6	8.3	42.8	20	14.3	0	7.1
Pop-Art	0	0	7.1	13.3	0	0	14.3
Expressionism	0	8.3	7.1	6.6	36	14.3	7.1
Impressionism	6.6	25	14.3	13.3	21.4	71.4	14.3
Cubism	0	0	7.1	20	14.3	0	35.7

Table A.10: Generative BoW topic model using CSIFT

using CSIFT and OSIFT features, with average accuracy of 49% and 50.3% respectively. Table A.12 summarizes the overall results for all the experiments. Figure A.13 shows the accuracies for classifying each style using all the evaluated models.

As can be examined from the results, the Discriminative model with Semantic-level features achieved the highest accuracy followed by Discriminative BoW with OSIFT, Generative BoW with OSIFT, Generative BoW with CSIFT and Discriminative BoW CSIFT. Also it can be deduced from the results that both Discriminative and Generative BoW models achieved comparable accuracy, while Discriminative Semantic model outperforms both BoW models. These results are inline with our hypothesis that the Semantic-level information would be more suitable for the task of fine-art style classification. By examining the results we can notice that the Baroque style is always classified with the highest accuracy in all techniques. It is also interesting to notice that the Popart style is classified with accuracy over 70% in all the discriminative approaches while the generative approach performed poorly in that style. Also it is worth noting that the OSIFT features outperformed the CSIFT features in the discriminative case; however the difference is not significant in the generative case.

Confusion(%)	Baroque	Abstract	Renaissance	Pop-Art	Expressionism	Impressionism	Cubism
Baroque	75.5	0	14.3	0	3.6	10.7	7.1
Abstract	0	62.5	3.5	27.3	3.6	3.6	0
Renaissance	7.1	4.2	39.2	3.3	7.1	3.6	10.7
Pop-Art	0	8.3	0	28	3.6	0	7.1
Expressionism	7.1	0	17.8	14	36	3.6	10.7
Impressionism	10.2	25	10.7	10.2	32	68	21.4
Cubism	0	0	14.3	16.9	14.3	10.7	42.9

Table A.11: Generative BoW topic model using OSIFT

Model	Dis Semantic	Dis BoW CSIFT	Dis BoW OSIFT	Gen BoW CSIFT	Gen BoW OSIFT
Mean Accuracy(%)	65.4	48.47	56.7	49	50.3
Std	4.8	2.45	3.26	2.43	2.46

Table A.12: Generative BoW topic model using OSIFT

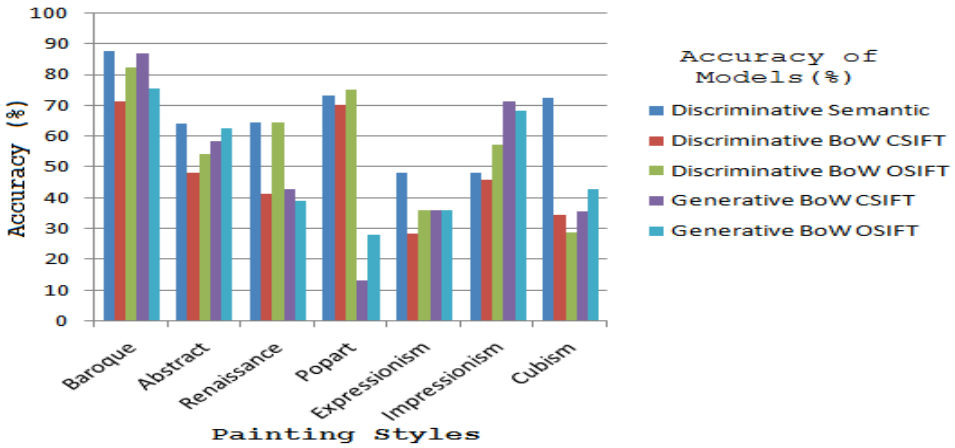

Figure A.13: Classification accuracy for each approach on each style

A.2.9 Influence Discovery Framework

Consider a set of artists, denoted by $A = \{a^l, l = 1 \cdots N_a\}$, where N_a is the number of artists. For each artist, a^l, we have a set of images of paintings, denoted by $P^l = \{p_i^l, i = 1, \cdots, N^l\}$, where N^l is the number of paintings for the l-th artist. For clarity of the presentation, we reserve the superscript for the artist index and the subscript for the painting index. We denote by $N = \sum_l N_l$ the total number of paintings. Following the conclusion of the style classification comparative study, we represent each painting by its Classeme features [165]. Therefore, each image $p_i^l \in R^D$ is a D dimensional feature vector that is the outcome of the Classeme classifiers, which defines the feature space.

To represent the temporal information, for each artist we have a ground truth time period where he/she performed their work, denoted by $t^l = [t^l_{start}, t^l_{end}]$ for the l-th artist, where t^l_{start} and t^l_{end} are the start and end year of that time period respectively. We do not consider the date of a given painting since for some paintings the exact time is unknown.

Painting Similarity:

To encode similarity/dissimilarity between paintings, we consider two different distances:

Euclidean distance: The distance $d_E(p_i^l, p_j^k)$ is defined to be the Euclidean distance between the Classeme feature vectors of paintings p_i^l and p_j^k. Since Classeme features are high-level semantic features, the Euclidean distance in the feature space is expected to measure dissimilarity in the subject matter between paintings. Painting similarity based on the Classeme features showed some interesting cases, several of which have not been studied before by art historians as a potential comparison. Figure A.9 is an example of this, as well as Figure A.14 and Figure A.15.

Manifold distance: Since the paintings in the feature space are expected to lie on a low-dimensional manifold, the Euclidean distance might be misleading in judging similarity/dissimilarity. Therefore, we also consider a manifold-based distance, $d_M(p_i^l, p_j^k)$ denoting the geodesic distance along the manifold of paintings in the feature space. To define such a distance, we use a method similar to ISOMAP [159], where we build a k-nearest neighbor graph of paintings, and compute the shortest path between each pair of paintings p_i^l and p_j^k on that graph. The distance $d_M(p_i^l, p_j^k)$ is then defined as the sum of the distances along the shortest path.

Figure A.14: Vincent van Gogh's *Old Vineyard with Peasant Woman* 1890 (left) and Joan Miro's *The Farm* 1922 (Right). Similar objects and scenery but different moods and style.

Artist Similarity:

Once painting similarity is encoded, using any of the two methods mentioned above, we can design a suitable similarity measure between artist. There are two challenges to achieve this task. First, how to define a measure of similarity between two artists, given their sets of paintings. We need to define a proper set distance $D(P^l, P^k)$ to encode the distance between the work of the l-th and k-th artists. This relates to how to define influence between artists to start with, where there is no clear definition. Should we declare an influence if one paining of artist k has strong similarity to a painting of artist l ? or if a number of paintings have similarity ? and what that "number" should be ?

Mathematically speaking, for a given painting $p_i^l \in P^l$ we can find its closest painting in P^k using a point-set distance as

$$d(p_i^l, P^k) = \min_j d(p_i^l, p_j^k).$$

We can find one painting in by artist l that is very similar to a painting by artist k, that can be considered an influence. This dictates defining an asymmetric distance measure in the form of

$$D_{min}(P^l, P^k) = \min_i d(p_i^l, P^k).$$

We denote this measure by *minimum-link influence*.

On the other hand, we can consider a central tendency in measuring influence, where we can measure the average or median of painting distances between P^l and P^k, we denote this measure *central-link influence*.

Figure A.15: Georges Braque's *Man with a Violin* 1912 (Left) and Pablo Picasso's *Spanish Still Life: Sun and Shadow* 1912 (Right).

Alternatively, we can think of Hausdorff distance [44], which measures the distance between two sets as the supremum of the point-set distances, defined as

$$D_H(P^l, P^k) = \max(\max_i d(p_i^l, P^k), \max_j d(p_j^k, P^l)).$$

We denote this measure *maximum-link influence*. Hausdorff distance is widely used in matching spatial points, which unlike a minimum distance, captures the configuration of all the points. While the intuition of Hausdorff distance is clear from a geometrical point of view, it is not clear what it means in the context of artist influence, where each point represent a painting. In this context, Hausdorff distance measures the maximum distance between any painting and its closest painting in the other set.

The discussion above highlights the challenge in defining the similarity between artists, where each of the suggested distance is in fact meaningful, and captures some aspects of similarity, and hence influence. In this paper, we do not take a position in favor of any of these measures, instead we propose to use a measure that can vary through the whole spectrum of distances between two sets of paintings. We define asymmetric distance between artist l and

artist k as the q-percentile Hausdorff distance, as

$$D_{q\%}(P^l, P^k) = \max_i^{q\%} d(p_i^l, P^k). \tag{A.2}$$

Varying the percentile q allows us to evaluate different settings ranging from a minimum distance, D_{min}, to a central tendency, to a maximum distance as in Hausdorff distance D_H.

Artist Influence Graph:

The artist asymmetric distance is used, in conjunction with the ground-truth time period to construct an influenced-by graph. The influence graph is a directed graph where each artist is represented by a node. A weighted directed edge between node i and node j indicates that artist i is potentially influenced by artist j, which is only possible if artist i succeed or is contemporary to artist j. The weight corresponds to the artist distance, i.e., a smaller weight indicates a higher potential influence. Therefore, the graph weights are defined as

$$w_{ij} = \begin{cases} D_{q\%}(P^i, P^j) & \text{if } t_{end}^i \geq t_{start}^j \\ \infty & \text{otherwise} \end{cases} \tag{A.3}$$

A.2.10 Influence Discovery Results

A.2.11 Evaluation Methodology:

We researched known influences between artists within our dataset from multiple resources such as *The Art Story Foundation* and *The Metropolitan Museum of Art*. For example, there is a general consensus among art historians that Paul Cézanne's use of fragmented spaces had a large impact on Pablo Picasso's work. In total, we collected 76 pairs of one-directional artist influences, where a pair (a^i, a^j) indicates that artist i is influenced by artist j. Figure A.16 shows the complete list of influenced-by list. Generally, it is a sparse list that contains only the influences which are consensual among many. Some artists do not have any influences in our collection while others may have up to five. We use this list as ground-truth for measuring the accuracy in our experiments.

The constructed influenced-by graph is used to retrieve the top-k potential influences for each artist. If a retrieved influence pair concur with an influence ground-truth pair, this is considered a hit. The hits are used to compute the recall, which is defined as the ratio between

Artist	Influenced by:				
BAZILLE	MANET	MONET	RENOIR	SISLEY	DELACROIX
BELLINI	MANTEGNA				
BLAKE	RAPHAEL	MICHELANGELO			
BOTTICELLI					
BRAQUE	PICASSO	CEZANNE			
BACON	PICASSO	VELAZQUEZ	VAN_GOGH	REMBRANDT	
BECKMANN	CEZANNE	MUNCH			
CAILLEBOTTE	DEGAS	MONET			
CAMPIN					
CARAVAGGIO					
CEZANNE	PISSARRO				
CHAGALL	PICASSO				
DEGAS	VELAZQUEZ	DELACROIX			
DELACROIX	MICHELANGELO	RUBENS	EL_GRECO		
DELAUNAY					
DONATELLO	GHIBERTI				
DURER	BELLINI	MANTEGNA			
EL_GRECO	TITIAN	MICHELANGELO			
GERICAULT	MICHELANGELO	RUBENS			
GHIBERTI					
GOYA					
GRIS					
HEPWORTH					
HOCKNEY	PICASSO				
HOFMANN	PICASSO	BRAQUE			
INGRES	RAPHAEL				
JOHNS					
KAHLO	BOTTICELLI				
KANDINSKY	CEZANNE	MONET	MARC		
KIRCHNER	DURER				
KLIMT	PICASSO	BRAQUE	DELAUNAY		
KLINE					
KLEE					
LEONARDO					
LICHTENSTEIN	PICASSO	JOHNS			
MACKE	Munch	DELAUNAY			
MALEVICH	CEZANNE				
MANET	VELAZQUEZ	MORISOT			
MANTEGNA	DONATELLO				
MARC	VAN_GOGH	DELAUNAY	KANDINSKY		
MICHELANGELO	GHIBERTI				
MONDRIAN	VAN_GOGH				
MONET					
MORISOT	MANET				
MOTHERWELL					
MIRO	CEZANNE	CHAGALL	VAN_GOGH		
MUNCH	MANET				
OKEEFFE					
PISSARRO					
PICASSO	EL_GRECO	GOYA			
RAPHAEL					
REMBRANDT					
RENOIR	MANET	DELACROIX			
RICHTER					
RODIN	MICHELANGELO				
ROUSSEAU					
RUBENS	MICHELANGELO	TITIAN			
Rothko					
SISLEY					
TITIAN	BELLINI				
VAN_EYCK					
VAN_GOGH	MONET	PISSARRO			
VELAZQUEZ	TITIAN	CARAVAGGIO			
VERMEER	CARAVAGGIO				
WARHOL	JOHNS				
ROCKWELL					

Figure A.16: Ground-truth artistic influences

Table A.13: Performance of influence retrieval using Euclidean distance and Classemes features.

	top-k recall				
q%	5	10	15	20	25
1	25	47.4	75	81.6	88.2
10	26.3	54	73.7	81.6	85.5
50	29	55.3	71.1	80.3	84.2
90	21.1	52.6	68.4	75	79
99	23.7	47.4	61.8	68.4	76.3

the correct influence detected and the total known influences in the ground truth. The recall is used for the sake of comparing the different settings relatively. Since detected influences can be correct although not in our ground truth, so there is no meaning to compute the precision.

A.2.12 Influence Discovery Validation

We experimented with the Classeme features, which showed the best results in the style classification task. We also experimented with GIST descriptors [116] and HOG descriptors [36], since they are the main ingredients in the Classemes features. In all cases, we computed the recall figures using the influence graph for the top-k similar artist (k=5, 10, 15, 25) with different q-percentile for the artist distance measure in Eq A.2 (q=1, 10, 50, 90, 99%). For all descriptors, we computed the influences using both the Euclidean distance and the Manifold-based distances. The results are shown in Tables A.2.12- A.2.12. The rows of the tables show different q-percentile. The columns show the recall percentage for the top-k similar artists . From the difference results we can see that most of the time the 50%-set distance (central-link influence) gives better results. We can also notice that generally the manifold-based distance slightly out performs the Euclidean distance for the same feature. Figure A.17 shows the recall curves using the Classemes features with different $q\%$. Figure A.18 compares the recall curves for different features (Classemes, GIST, HOG) and distances (Euclidean vs Manifolds), all calculated using the 50% set distance. The results using the three features seems to be comparable.

Table A.14: Performance of influence retrieval using manifold-based distance and Classemes features.

	top-k recall				
q%	5	10	15	20	25
1	25	50	73.7	85.5	89.5
10	27.6	61.8	75	83	90.8
50	31.6	57.9	71.1	80.3	84.2
90	26.3	51.3	68.4	77.6	84.2
99	21.1	47.4	67.1	75	81.6

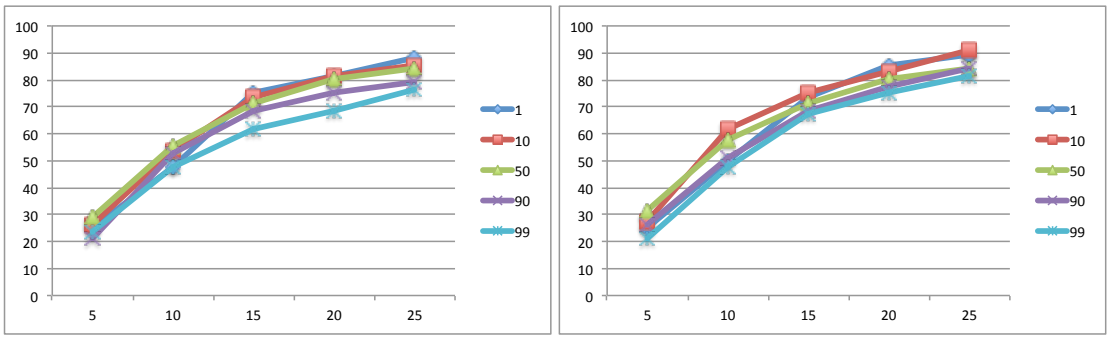

Figure A.17: Influence recall curves, using classemes features with different q%. Left: Euclidean distance, Right: Manifold distance.

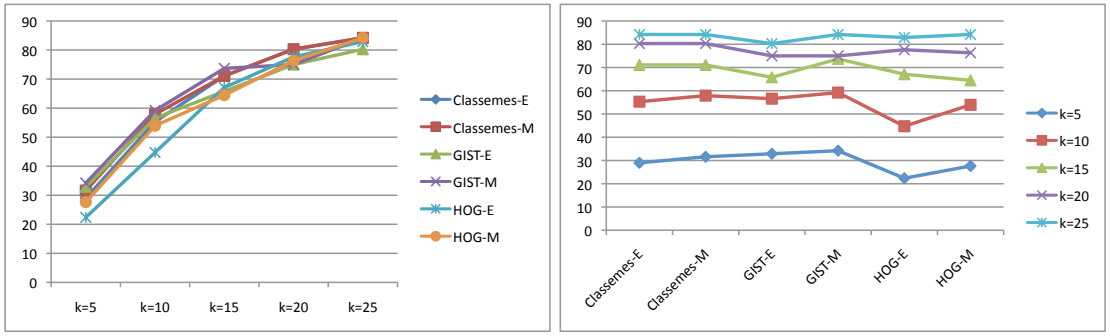

Figure A.18: Influence Recall at different top-k: Comparisons of different descriptors

Table A.15: Performance of influence retrieval using Euclidean distance and GIST features.

	top-k recall				
q%	5	10	15	20	25
1	21.05	40.79	60.53	69.74	75.00
10	31.58	50.00	65.79	71.05	76.32
50	32.89	56.58	65.79	75.00	80.26
90	28.95	55.26	72.37	76.32	84.21
99	23.68	48.68	68.42	76.32	81.58

Table A.16: Performance of influence retrieval using manifold-based distance and GIST features.

	top-k recall				
q%	5	10	15	20	25
1	22.37	42.11	63.16	68.42	73.68
10	34.21	53.95	67.11	69.74	78.95
50	34.21	59.21	73.68	75.00	84.21
90	30.26	55.26	71.05	73.68	78.95
99	21.05	48.68	67.11	73.68	81.58

Table A.17: Performance of influence retrieval using Euclidean distance and HOG features.

	top-k recall				
q%	5	10	15	20	25
1	22.37	40.79	56.58	71.05	78.95
10	22.37	47.37	64.47	78.95	82.89
50	22.37	44.74	67.11	77.63	82.89
90	25.00	52.63	67.11	77.63	84.21
99	26.32	48.68	63.16	73.68	78.95

Table A.18: Performance of influence retrieval using manifold-based distance and HOG features.

	top-k recall				
q%	5	10	15	20	25
1	23.68	39.47	57.89	71.05	80.26
10	25.00	46.05	63.16	76.32	80.26
50	27.63	53.95	64.47	76.32	84.21
90	23.68	46.05	65.79	75.00	81.58
99	27.63	43.42	57.89	68.42	71.05

A.2.13 Visualizing Influences - A Map of Artists

The influence graph can be used to achieve a visualization of artists and their similarities, i.e. a Map of Artists. For this purpose we used ISOMAP [159] to achieve a low-dimensional embedding of the artist influence graph. ISOMAP computes the shortest path on the graph between each two artists, and use that to achieve an embedding using multi-dimensional scaling (MDS) [22]. The reason we use ISOMAP in particular, among several other low-dimensional embedding techniques, is that ISOMAP works with directed graphs.

Figure A.19 and A.20 illustrate a visualization of artist similarity based on embedding the influence graph into a three-dimensional space using ISOMAP, each plot shows a two-dimensional projection of that space. The artists are color coded in these plots to reflect their ground-truth style. We can see that artist of the same style are mostly clustered together. For example a few *Expressionist* artists clustered together as well as *Abstract Contemporary* artists. As seen, the artists populated the right of the mapping are Lichtenstein, Hepworth, Malevich, Mondrian, Motherwell, O'Keffe, and Rothko, who are all Modern and Abstract artists. Their styles differ slightly but all share some stylistic approaches and time period. On the left side of the plot we can find most *Impressionists* and *Renaissance* artists. However, we can see that the *Impressionists* and *Renaissance* artists seem to have similar values in one dimension but not the other. It is also clear that the distances within and between the *Impressionists* and *Renaissance* (in the right side) are much smaller than the distances among the *Expressionist* and *Abstract Contemporary* artists (in the left side). Other styles, such as *Romanticism*, seem to have a broader range of values.

Some artists in this mapping seem to cluster according to their style, but in the context of influence, it is also important to think about the similarities between artists instead of the classification of style. This is yet another complication of the task of measuring influence. Therefore, another way to analyze this graph is to disregard style all together. We can wonder whether Richter and Hockney share a connection because they lie close to each other. Or we can wonder if Klimt was influenced by Picasso or Braque. In fact, both Picasso and Braque were listed as influences for Klimt in our ground-truth list. When comparing these close mappings to the ground truth influence, some are reasonable while others seem less coherent. In another

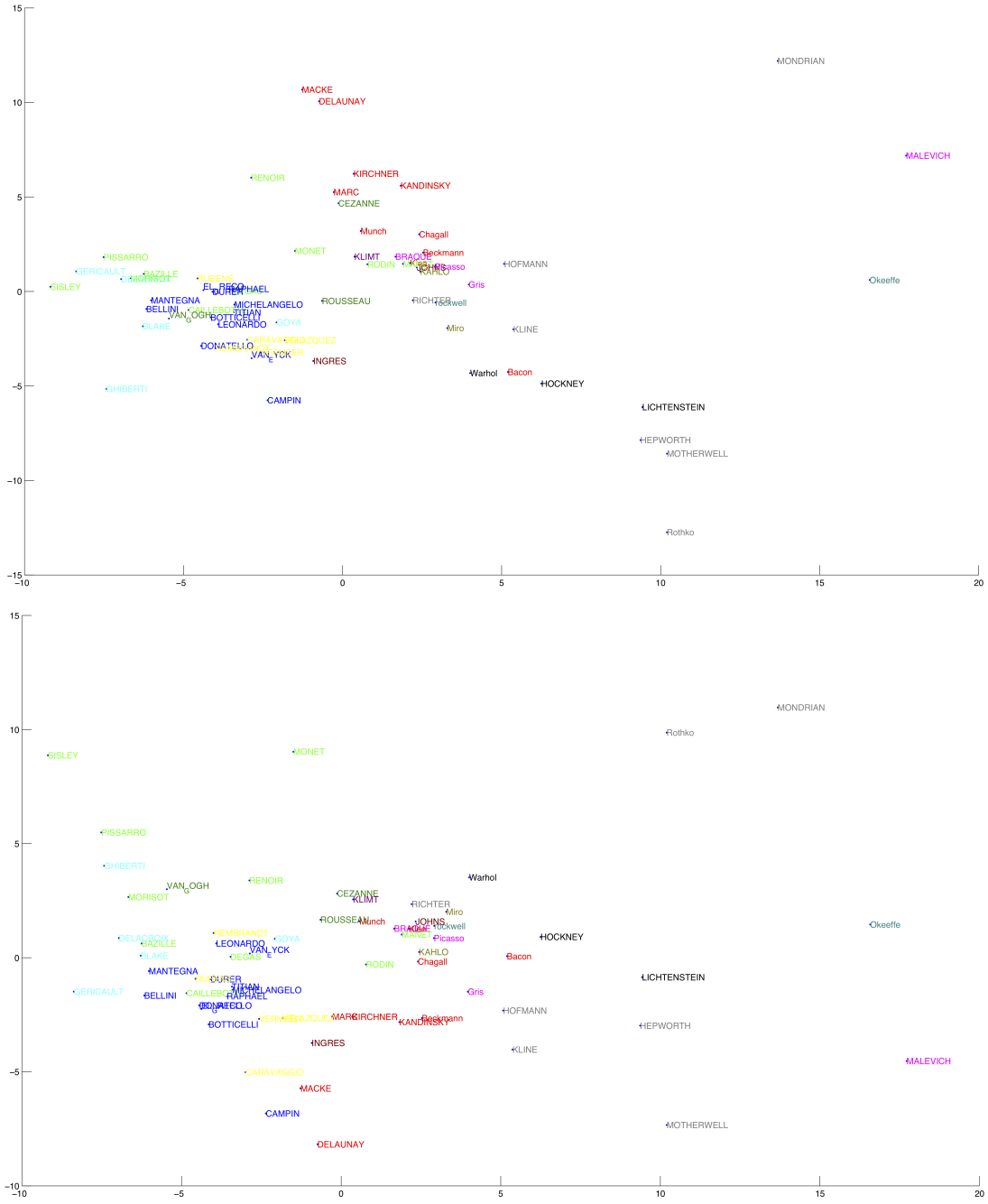

Figure A.19: Map of Artists: Similar artists in two dimensions: Top: Dimensions 1 and 2, Bottom: Dimensions 1 and 3. Artist are color coded by their style.

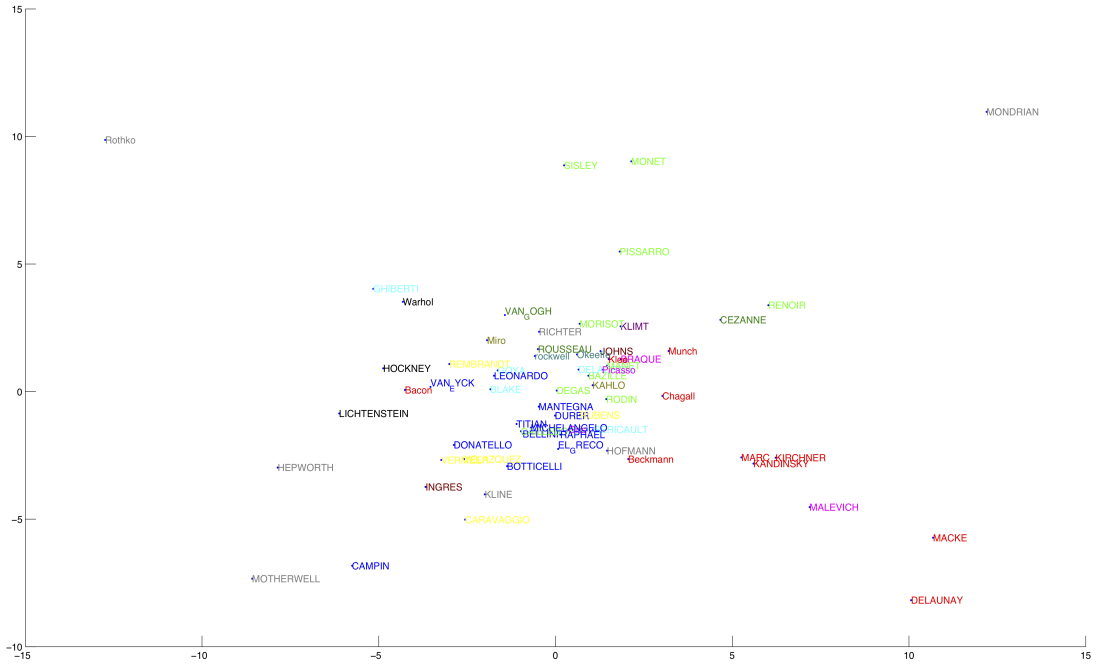

Figure A.20: Map of Artists: Similar artists in two dimensions: Top: Dimensions 2 and 3. Artists are color coded by their style.

example, Bazille lies close to Delacroix which is consistent with our ground truth. Other successful mappings include Munch's influence on Beckmann, Degas's influence on Caillebotte, and others. Figure A.21 illustrates the top-5 suggested influence results.

A.2.14 Conclusion and Future Works

This chapter scratches the surface of the problem of automated discovery of artist influence, through the study of painting and artist similarity. We posed the interesting question of finding influence between painters as a knowledge discovery problem and showed interesting results for both of the qualitative and quantitative measurements.

In this chapter we also studied the problem of paintings style classification, and presented a comparative study of three different models for the classification task, with different visual features. That study showed that semantic-level features perform the best for this task. This conclusion lead us to use these semantic features for the task of influence discovery.

Artist	Influenced By:				
'BAZILLE'	'CEZANNE'	'RUBENS'	'DURER'	'DELACROIX'	'DEGAS'
'BELLINI'	'RAPHAEL'	'MANTEGNA'	'BOTTICELLI'	'VAN_EYCK'	'TITIAN'
'BLAKE'	'EL_GRECO'	'RAPHAEL'	'DURER'	'DELACROIX'	'VELAZQUEZ'
'BOTTICELLI'	'MANTEGNA'	'VAN_EYCK'	'MICHELANGE	'DURER'	'BELLINI'
'BRAQUE'	'Picasso'	'CEZANNE'	'RAPHAEL'	'JOHNS'	'MANTEGNA'
'Bacon'	'MANET'	'Beckmann'	'Picasso'	'VELAZQUEZ'	'RAPHAEL'
'Beckmann'	'Picasso'	'DURER'	'VELAZQUEZ'	'RAPHAEL'	'MICHELANGELO'
'CAILLEBOTTE	'MANET'	'DELACROIX'	'CEZANNE'	'VAN_EYCK'	'DURER'
'CAMPIN'	'DONATELLO'	[]	[]	[]	[]
'CARAVAGGIO	'RUBENS'	'TITIAN'	'EL_GRECO'	'LEONARDO'	'RAPHAEL'
'CEZANNE'	'Picasso'	'RENOIR'	'RUBENS'	'DELACROIX'	'EL_GRECO'
'Chagall'	'RAPHAEL'	'Picasso'	'Beckmann'	'MICHELANGE	'DELACROIX'
'DEGAS'	'CEZANNE'	'Picasso'	'RAPHAEL'	'DELACROIX'	'Munch'
'DELACROIX'	'RUBENS'	'EL_GRECO'	'RAPHAEL'	'DURER'	'TITIAN'
'DELAUNAY'	'MARC'	'Beckmann'	'MALEVICH'	'MACKE'	'CEZANNE'
'DONATELLO'	'MANTEGNA'	'VAN_EYCK'	'LEONARDO'	'BOTTICELLI'	'CAMPIN'
'DURER'	'LEONARDO'	'MANTEGNA'	'VAN_EYCK'	'RAPHAEL'	'TITIAN'
'EL_GRECO'	'RUBENS'	'TITIAN'	'DURER'	'RAPHAEL'	'MANTEGNA'
'GERICAULT'	'DELACROIX'	'TITIAN'	'RUBENS'	'GOYA'	'RAPHAEL'
'GHIBERTI'	'VAN_EYCK'	'DONATELLO'	'MANTEGNA'	'CAMPIN'	[]
'GOYA'	'REMBRANDT'	'LEONARDO'	'DELACROIX'	'TITIAN'	'VELAZQUEZ'
'Gris'	'Picasso'	'Miro'	'BRAQUE'	'JOHNS'	'Munch'
'HEPWORTH'	'Picasso'	'Gris'	'JOHNS'	'Bacon'	'KLINE'
'HOCKNEY'	'RAPHAEL'	'MANET'	'rockwell'	'HOFMANN'	'Picasso'
'HOFMANN'	'MALEVICH'	'Munch'	'Klee'	'KLINE'	'MACKE'
'INGRES'	'TITIAN'	'LEONARDO'	'CARAVAGGIO	'VELAZQUEZ'	'JOHNS'
'JOHNS'	'Picasso'	'BRAQUE'	'DURER'	'CEZANNE'	'RAPHAEL'
'KAHLO'	'CEZANNE'	'RAPHAEL'	'Picasso'	'RENOIR'	'LEONARDO'
'KANDINSKY'	'Chagall'	'BRAQUE'	'RAPHAEL'	'RUBENS'	'MICHELANGELO'
'KIRCHNER'	'EL_GRECO'	'Beckmann'	'Picasso'	'DELACROIX'	'MARC'
'KLIMT'	'VAN_EYCK'	'Picasso'	'JOHNS'	'Munch'	'MANTEGNA'
'KLINE'	'Beckmann'	'CAILLEBOTTE	'MANET'	'INGRES'	'VELAZQUEZ'
'Klee'	'Picasso'	'CEZANNE'	'VERMEER'	'KLIMT'	'JOHNS'
'LEONARDO'	'DURER'	'RAPHAEL'	'VAN_EYCK'	'MANTEGNA'	'TITIAN'
'LICHTENSTEI	'Picasso'	'HOFMANN'	'Beckmann'	'RODIN'	'ROUSSEAU'
'MACKE'	'RAPHAEL'	'RUBENS'	'MARC'	'Picasso'	'CEZANNE'
'MALEVICH'	'Gris'	'Miro'	'Picasso'	'VELAZQUEZ'	'KAHLO'
'MANET'	'VELAZQUEZ'	'CEZANNE'	'RAPHAEL'	'Picasso'	'DEGAS'
'MANTEGNA'	'BOTTICELLI'	'VAN_EYCK'	'DURER'	'LEONARDO'	'MICHELANGELO'
'MARC'	'EL_GRECO'	'RAPHAEL'	'KIRCHNER'	'MICHELANGE	'Picasso'
'MICHELANGE	'RAPHAEL'	'DURER'	'TITIAN'	'MANTEGNA'	'LEONARDO'
'MONDRIAN'	'MALEVICH'	'BRAQUE'	'Picasso'	'KLIMT'	'JOHNS'
'MONET'	'PISSARRO'	'CEZANNE'	'SISLEY'	'VAN_GOGH'	'RENOIR'
'MORISOT'	'CEZANNE'	'RENOIR'	'DELACROIX'	'PISSARRO'	'MONET'
'MOTHERWELL	'VELAZQUEZ'	'Beckmann'	'RODIN'	'Bacon'	'TITIAN'
'Miro'	'Picasso'	'Gris'	'JOHNS'	'VELAZQUEZ'	'ROUSSEAU'
'Munch'	'CEZANNE'	'Picasso'	'DURER'	'BRAQUE'	'MANTEGNA'
'Okeeffe'	'BRAQUE'	'MALEVICH'	'MONDRIAN'	'VAN_EYCK'	'MICHELANGELO'
'PISSARRO'	'CEZANNE'	'MONET'	'VAN_GOGH'	'RENOIR'	'SISLEY'
'Picasso'	'BRAQUE'	'CEZANNE'	'DURER'	'RAPHAEL'	'EL_GRECO'
'RAPHAEL'	'MICHELANGE	'DURER'	'MANTEGNA'	'TITIAN'	'LEONARDO'
'REMBRANDT'	'LEONARDO'	'TITIAN'	'VELAZQUEZ'	'DURER'	'EL_GRECO'
'RENOIR'	'CEZANNE'	'DEGAS'	'RUBENS'	'TITIAN'	'RAPHAEL'
'RICHTER'	'GOYA'	'RUBENS'	'TITIAN'	'RODIN'	'LEONARDO'
'RODIN'	'CEZANNE'	'VELAZQUEZ'	'DELACROIX'	'EL_GRECO'	'TITIAN'
'ROUSSEAU'	'VAN_GOGH'	'VAN_EYCK'	'BOTTICELLI'	'Picasso'	'DURER'
'RUBENS'	'TITIAN'	'EL_GRECO'	'RAPHAEL'	'VELAZQUEZ'	'DURER'
'Rothko'	'HOCKNEY'	'VELAZQUEZ'	'Picasso'	'Bacon'	'BRAQUE'
'SISLEY'	'PISSARRO'	'CEZANNE'	'MONET'	'RENOIR'	'VAN_GOGH'
'TITIAN'	'RAPHAEL'	'EL_GRECO'	'DURER'	'LEONARDO'	'BOTTICELLI'
'VAN_EYCK'	'DONATELLO'	'CAMPIN'	[]	[]	[]
'VAN_GOGH'	'CEZANNE'	'MANTEGNA'	'DELACROIX'	'PISSARRO'	'DURER'
'VELAZQUEZ'	'EL_GRECO'	'RAPHAEL'	'LEONARDO'	'REMBRANDT'	'TITIAN'
'VERMEER'	'LEONARDO'	'VELAZQUEZ'	'VAN_EYCK'	'REMBRANDT'	'DURER'
'Warhol'	'Bacon'	'rockwell'	'Beckmann'	'LEONARDO'	'DEGAS'
'rockwell'	'Picasso'	'RAPHAEL'	'CEZANNE'	'DELACROIX'	'BRAQUE'

Figure A.21: Top-5 suggested influences retrieved from the graph: using Classemes features, Euclidean distance, and q=50%,

For the task of influence discovery, we compared several distance measures between paintings, including a Euclidean distance and a manifold-based distance. The comparative experiments showed that the manifold-based distance gave slightly better results. We proposed and evaluated different artist distance measures, denoted as minimum-link, central-link, and maximum-link influence measures. This problem can be formulated as a set distance, however the typical Hausdorff set distance did not perform best, instead the central-link influence measure performed best in all experiments. We also present a tool for visualizing artist similarity through what we call a map of artists.

In this chapter we also presented a new annotated dataset with diverse set of artists and wide range of paintings. This dataset will be publicly available and can be used for interdisciplinary tasks of Art and Computer Science.

Of course, there is a lot more to be done. For example, our framework could include searching for specific stylistic similarities such as brushstroke and pattern. We could also include more features of color and line. We can experiment with many other features especially among the elements and principles of art. Clearly there are many ways in which artists are influenced by each other. This is why mapping influence is such a difficult task.

A.3 Quantifying Creativity in Art Networks

1. Ahmed Elgammal, Babak Saleh: Quantifying Creativity in Art Networks, International Conference on Computational Creativity (ICCC) 2015.

Can we develop a computer algorithm that assesses the creativity of a painting given its context within art history? This paper proposes a novel computational framework for assessing the creativity of creative products, such as paintings, sculptures, poetry, etc. We use the most common definition of creativity, which emphasizes the originality of the product and its influential value. The proposed computational framework is based on constructing a network between creative products and using this network to infer about the originality and influence of its nodes. Through a series of transformations, we construct a Creativity Implication Network. We show that inference about creativity in this network reduces to a variant of network centrality problems which can be solved efficiently. We apply the proposed framework to the task of quantifying creativity of paintings (and sculptures). We experimented on two datasets with over 62K paintings to illustrate the behavior of the proposed framework. We also propose a methodology for quantitatively validating the results of the proposed algorithm, which we call the "time machine experiment".

A.3.1 Introduction

The field of computational creativity is focused on giving the machine the ability to generate human-level "creative" products such as computer generated poetry, stories, jokes, music, art, etc., as well as creative problem solving. An important characteristic of a creative agent is its ability to assess its creativity as well as judge other agents' creativity. In this paper we focus on developing a computational framework for assessing the creativity of products, such as painting, sculpture, etc. We use the most common definition of creativity, which emphasizes the originality of the product and its influential value [124]. In the next section we justify the use of this definition in contrast to other definitions. The proposed computational framework is based on constructing a network between products and using it to infer about the originality and influence of its nodes. Through a series of transformations, we show that the problem can reduce to a variant of network centrality problems, which can be solved efficiently.

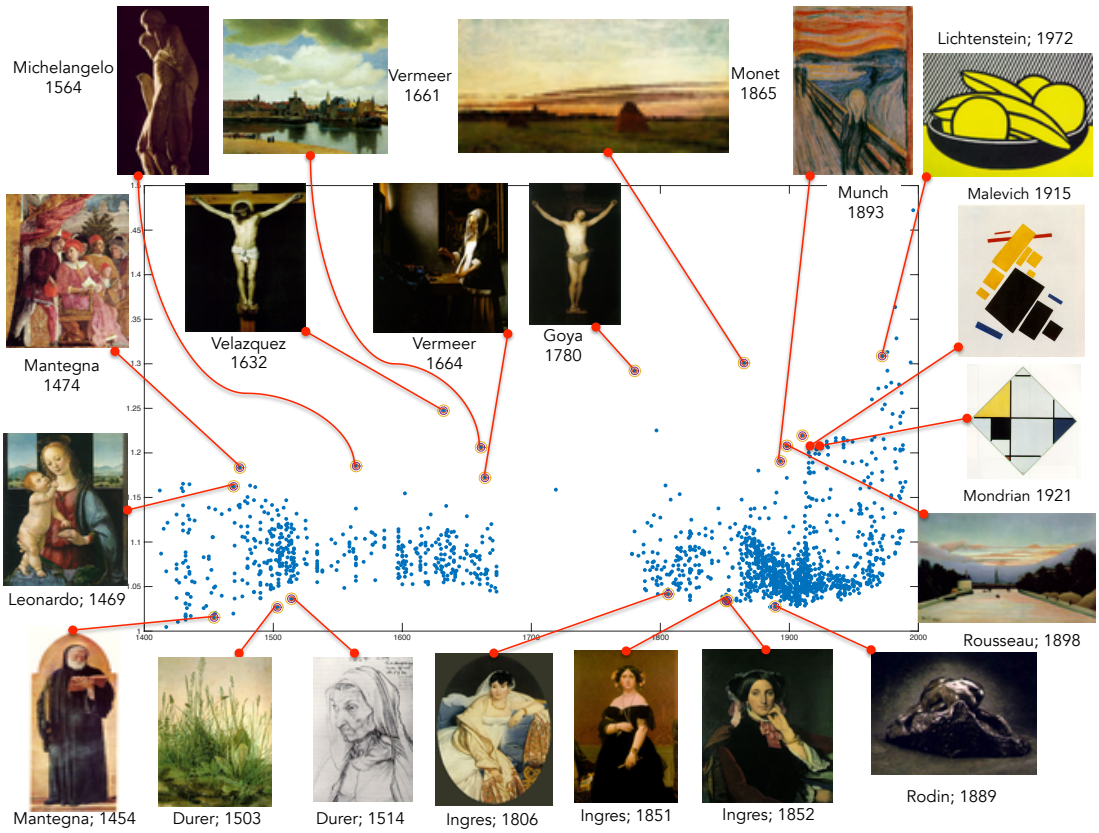

Figure A.22: Creativity scores for 1710 paintings from Artchive dataset. Each point represents a painting. The horizontal axis is the year the painting was created and the vertical axis is the creativity score (scaled). The thumbnails illustrate some of the painting that scored relatively high or low compared to their neighbors. Only artist names and dates of the paintings are shown on the graph because of limited space. See Figure A.24 for a zoom in to the period 1850-1950

We apply the proposed framework to the task of quantifying creativity of paintings (and sculptures). The reader might question the feasibility, limitation, and usefulness of performing such task by a machine. Artists, art historians and critics use different concepts to describe pantings. In particular, elements of arts such as space, texture, form, shape, color, tone and line. Artists also use principles of art including movement, unity, harmony, variety, balance, contrast, proportion, and pattern; besides brush strokes, subject matter, and other descriptive concepts [60]. We collectively call these concepts artistic concepts. These artistic concepts can, more or less, be quantified by today's computer vision technology. With the rapid progress in computer vision, more advanced techniques are introduced, which can be used to measure similarity between paintings with respect to a given artistic concept. Whether the state of the art is already sufficient to measure similarity in meaningful ways, or whether this will

happen in the near or far future, the goal of this paper is to design a framework that can use such similarity measures to quantify our chosen definition of creativity in an objective way. Hence, the proposed framework would provide a ready-to-use approach that can utilize any future advances in computer vision that might provide better ways for visual quantification of digitized paintings. In fact, we applied the proposed framework using state-of-the-art computer vision techniques and achieved very reasonable automatic quantification of creativity on two large datasets of paintings. Figure A.22 illustrates an example of the creativity scores obtained on dataset containing 1710 paintings.

One of the fundamental issues with the problem of quantifying creativity of art is how to validate any results that the algorithm can obtain. Even if art historians would agree on a list of highly original and influential paintings that can be used for validation, any algorithm that aims at assigning creativity scores will encounter three major limitations: I) Closed-world limitation: The algorithm is only limited to the set of paintings it analyzed. It is a closed world for the algorithm where this set is every thing it has seen about art history. The number of images of paintings available in the public domain is just a small fraction of what are in museums and private collections. II) Artistic concept quantification limitation: the algorithm is limited by what it sees, in terms of the ability of the underlying computer vision methods to encode the important elements and principles of art that relates to judging creativity. III) Parameter setting: the results will depend on the setting of the parameters, where each setting would mean a different way to assign creativity scores with different interpretation and different criteria. However, these limitations should not stop us from developing and testing algorithms to quantify creativity. The first two limitations are bound to disappear in the future, with more and more paintings being digitized, as well as with the continuing advances in computer vision and machine learning. The third limitation should be thought of as an advantage, since the different settings mean a rich ability of the algorithm to assign creativity scores based on different criteria. For the purpose of validation, we propose a methodology for validating the results of the algorithm through what we denote as “time machine experiments”, which provides evidence of the correctness of the algorithm.

Having discussed the feasibility and limitations, let us discuss the value of using any computational framework to assess creativity in art. For a detailed discussion about the implications

of using computational methods in the domain of aesthetic-judgment-related tasks, we refer the reader to [155]. Our goal is not to replace art historians' or artists' role in judging creativity of art products. Providing a computational tool that can process millions of artworks to provide objective similarity measures and assessments of creativity, given certain visual criteria can be useful in the age of digital humanities. From a computational creativity point of view, evaluating the framework on digitized art data provides an excellent way to optimize and validate the framework, since art history provides us with suggestions about what is considered creative and what might be less creative. In this work we did not use any such hints in achieving the creativity scores, since the whole process is unsupervised, i.e., the approach does not use any creativity, genre, or style labels. However we can use evidence from art history to judge whether the results make sense or not. Validating the framework on digitized art data makes it possible to be used on other products where no such knowledge is available, for example to validate computer-generated creative products.

A.3.2 On the Notion of Creativity

There is a historically long and ongoing debate on how to define creativity. In this section we give a brief description of some of these definitions that directly relate to the notion we will use in the proposed computational framework. Therefore, this section is by no means intended to serve as a comprehensive overview of the subject. We refer readers to [158, 124] for comprehensive overviews of the different definitions of creativity.

We can describe a person (e.g. artist, poet), a product (painting, poem), or the mental process as being creative [158, 124]. Among the various definitions of creativity it seems that there is a convergence to two main conditions for a product to be called "creative". That product must be novel, compared to prior work, and also has to be of value or influential [124]. These criteria resonate with Kant's definition of artistic genius, which emphasizes two conditions "originality" and being "exemplary" [7]. Psychologists would not totally agree with this definition since

[7] Among four criteria for artistic genius suggested by Kant, two describe the characteristic of a creative product "That genius 1) is a talent for producing that for which no determinate rule can be given, not a predisposition of skill for that which can be learned in accordance with some rule, consequently that originality must be it's primary characteristic. 2) that since there can also be original nonsense, its products must at the same time be models, i.e., exemplary, hence, while not themselves the result of imitation, they must yet serve others in that way, i.e., as a standard or rule for judging." [71]-p186

they favor associating creativity with the mental process that generates the product [158, 111]. However associating creativity with products makes it possible to argue in favor of "Computational Creativity", since otherwise, any computer product would be an output of an algorithmic process and not a result of a creative process. Hence, in this paper we stick to quantifying the creativity of products instead of the mental process that create the product.

Boden suggested a distinction between two notions of creativity: psychological creativity (P-creativity), which assesses novelty of ideas with respect to its creator, and historical creativity (H-creativity), which assesses novelty with respect to the whole human history [21]. It follows that P-creativity is a necessary but not sufficient condition for H-creativity, while H-creativity implies P-creativity [21, 111]. This distinction is related to the subjective (related to person) vs. objective creativity (related to the product) suggested by Jarvie [79]. In this paper our definition of creativity is aligned with objective/H-creativity, since we mainly quantify creativity within a historical context.

A.3.3 Computational Framework

According to the discussion in the previous section, a creative product must be *original*, compared to prior work, and valuable (*influential*) moving forward. Let us construct a network of creative products and use it to assign a creativity score to each product in the network according to the aforementioned criteria. In this section, for simplicity and without loss of generality, we describe the approach based on a network of paintings, however the framework is applicable to other art or literature forms.

A.3.4 Constructing a Painting Graph

Let us denote by $P = \{p_i, i = 1 \cdots N\}$ a set of paintings. The goal is to assign a creativity score for each painting, denoted by $C(p_i)$ for painting p_i . Every painting comes with a time label indicating the date it was created, denoted by $t(p_i)$. We create a directed graph where each vertex corresponds to a painting. A directed edge (arc) connects painting p_i to p_j if p_i was created before p_j. Each directed edge is assigned a positive weight (we will discuss later where the weights come from), we denote the weight of edge (p_i, p_j) by w_{ij}. We denote by W_{ij} the adjacency matrix of the painting graph, where $W_{ij} = w_{ij}$ if there is an edge from p_i to p_j and

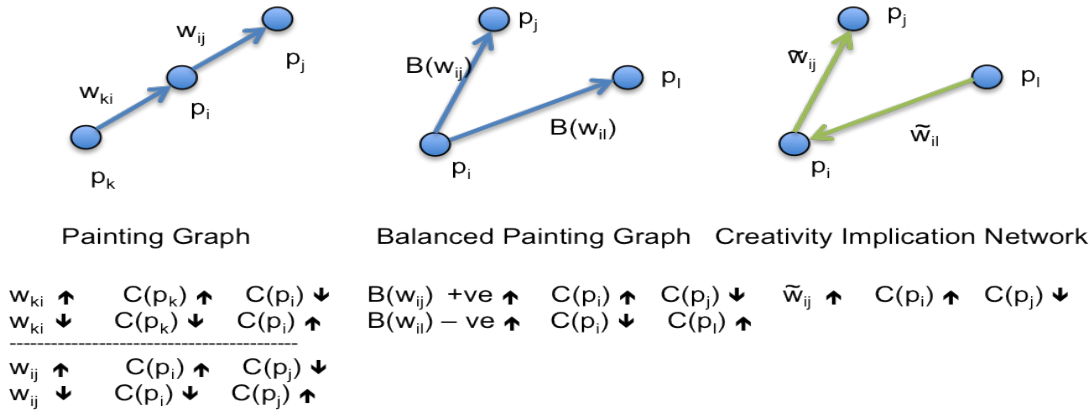

Figure A.23: Illustration of the construction of the Creativity Implication Network: blue arrows indicate temporal relation and orange arrows indicate reverse creativity implication (converse).

0 otherwise. Note that according to this definition, a painting is not connected to itself, i.e., $w_{ii} = 0, i = 1 \cdots N$. By construction, $w_{ij} > 0 \rightarrow w_{ji} = 0$, i.e., the graph is anti-symmetric.

To assign the weights we assume that there is a similarity function that takes two paintings and produces a positive scalar measure of affinity between them (higher value indicates higher similarity). We denote such a function by $S(\cdot, \cdot)$ and, therefore,

$$w_{ij} = \begin{cases} S(p_i, p_j) & \text{if } t(p_i) < t(p_j). \\ 0 & \text{otherwise.} \end{cases}$$

Since there are multiple possible visual aspects that can be used to measure similarity, we denote such a function by $S^a(\cdot, \cdot)$ where the superscript a indicates the visual aspect that is used to measure the similarity (color, subject matter, brush stroke, etc.) This implies that we can construct multiple graphs, one for each similarity function. We denote the corresponding adjacency matrix by W^a, and the induced creativity score by C^a, which measure the creativity along the dimension of visual aspect a. In the rest of this section, for the sake of simplicity, we will assume one similarity function and drop the superscript. Details about the similarity function will be explained in the next section.

A.3.5 Creativity Propagation

Giving the constructed painting graph, how can we propagate the creativity in such a network? To answer this question we need to understand the implication of the weight of the directed

edge connecting two nodes on their creativity scores. Let us assume that initially we assign equal creativity indices to all nodes. Consider painting p_i and consider an incoming edge from a prior painting p_k. A high weight on that edge (w_{ki}) indicates a high similarity between p_i and p_k, which indicates that p_i is not novel, implying that we should lower the creativity score of p_i (since p_i is subsequent to p_k and similar to it) and increase the creativity score of p_k. In contrast, a low weight implies that p_i is novel and hence creative compared to p_k, therefore we need to increase the creativity score of p_i and decreases that of p_k.

Let us now consider the outgoing edges from p_i. According to our notion of creativity, for p_i to be creative it is not enough to be novel, it has to be influential as well (some others have to imitate it). This indicates that a high weight, w_{ij}, between p_i and a subsequent painting p_j implies that we should increase the creativity score of p_i and decrease that of p_j. In contrast, a lower weight implies that p_i is not influential on p_j, and hence we should decrease the score for p_i and increase it for p_j. These four cases are illustrated in Figure A.23. A careful look reveals that the two cases for the incoming edges and those for the outgoing edges are in fact the same. *A higher weight implies the prior node is more influential and the subsequent node is less creative, and a lower weight implies the prior node is less influential and the subsequent node is more creative.*

A.3.6 Creativity Implication Network

Before converting this intuition to a computational approach, we need to define what is considered high and low for weights. We introduce a balancing function on the graph. Let $m(i)$ denote a balancing value for node i, where for the edges connected to that node a weight above $m(i)$ is considered high and below that value is considered low. We define a balancing function as a linear function on the weights connecting to each node in the form

$$B_i(w) = \begin{cases} w - m(i) & \text{if } \ w > 0. \\ 0 & \text{otherwise.} \end{cases}$$

We can think of different forms of balancing functions that can be used. Also there are different ways to set the parameter $m(i)$ with different implications, which we will discuss in the next section. This form of balancing function basically converts weights lower than $m(i)$

to negative values. The more negative the weight of an edge the more creative the subsequent node and the less influential the prior node. The more positive the weight of an edge the less creative the subsequent node and the more influential the prior node.

The introduction of the negative weights in the graph, despite providing a solution to represent low weights, is problematic when propagating the creativity scores. The intuition is, a negative edge between p_i and p_j is equivalent to a positive edge between p_j and p_i. This directly suggests that we should reverse all negative edges and negate their values. Notice that the original graph construction guarantees that an edge between p_i and p_j implies no edge between p_j and p_i, therefore there is no problem with edge reversal. This process results in what we call *"Creativity Implication Network"*. We denote the weights of that graph by $\tilde{w}_{ij}$ and its adjacency matrix by $\widetilde{W}$. This process can be described mathematically as

$$\begin{aligned} B(w_{ij}) > 0 \quad &\rightarrow \quad \tilde{w}_{ij} = B(w_{ij}) \\ B(w_{ij}) = 0 \quad &\rightarrow \quad \tilde{w}_{ij} = 0 \\ B(w_{ij}) < 0 \quad &\rightarrow \quad \tilde{w}_{ji} = -B(w_{ij}) \end{aligned}$$

The Creativity Implication Network has one simple rule that relates its weights to creativity propagation: *the higher the weight of an edge between two nodes, the less creative the subsequent node and the more creative the prior node.* Note that the direction of the edges in this graph is no longer related to the temporal relation between its nodes, instead it is directly inverse to the way creativity scores should propagate from one painting to another. Notice that the weights of this graph are non-negative.

A.3.7 Computing Creativity Scores

Given the construction of the Creativity Implication Network, we are now ready to define a recursive formula for assigning creativity scores. We will show that the construction of the Creativity Implication Network reduces the problem of computing the creativity scores to a traditional network centrality problem. The algorithm will maintain creativity scores that sum up to one, i.e., the creativity scores form a probability distribution over all the paintings in our

set. Given an initial equal creativity scores, the creativity score of node p_i should be updated as

$$C(p_i) = \frac{(1-\alpha)}{N} + \alpha \sum_j \tilde{w}_{ij} \frac{C(p_j)}{N(p_j)}, \tag{A.4}$$

where $0 \leq \alpha \leq 1$ and $N(p_j) = \sum_k \tilde{w}_{kj}$. In this formula, the creativity of node p_i is computed from aggregating a fraction α of the creativity scores from its outgoing edges weighted by the adjusted weights $\tilde{w}_{ij}$. The constant term $(1-\alpha)/N$ reflects the chance that similarity between two paintings might not necessarily indicate that the subsequent one is influenced by the prior one. For example, two paintings might be similar simply because they follow a certain style or art movement. The factor $1-\alpha$ reflects the probability of this chance. The normalization term $N(p_j)$ for node j is the sum of its incoming weights, which means that the contribution of node p_j is split among all its incoming nodes based on the weights, and hence, p_i will collect only a fraction $\tilde{w}_{ij}/\sum_k \tilde{w}_{kj}$ of the creativity score of p_j.

The recursive formula in Eq A.4 can be written in a matrix form as

$$C = \frac{(1-\alpha)}{N}\mathbf{1} + \alpha \widetilde{\widetilde{W}} C, \tag{A.5}$$

where $\widetilde{\widetilde{W}}$ is a column stochastic matrix defined as $\widetilde{\widetilde{W}}_{ij} = \tilde{w}_{ij}/\sum_k \tilde{w}_{kj}$, and $\mathbf{1}$ is a vector of ones of the same size as C. It is easy to see that since $\widetilde{\widetilde{W}}$, C, and $\frac{1}{N}\mathbf{1}$ are all column stochastic, the resulting scores will always sum up to one. The creativity scores can be obtained by iterating over Eq A.5 until conversion. Also a closed-form solution for the case where $\alpha \neq 1$ can be obtained as

$$C^* = \frac{(1-\alpha)}{N}(I - \alpha \widetilde{\widetilde{W}})^{-1}\mathbf{1}. \tag{A.6}$$

A reader who is familiar with social network analysis literature might directly see the relation between this formulation and some traditional network centrality algorithms. Eq A.5 represents a random walk in a Markov chain. Setting $\alpha = 1$, the formula in Eq A.5 becomes a weighted variant to eigenvector centrality [23], where a solution can be obtained by the right eigenvector corresponding to the largest eigenvalue of $\widetilde{\widetilde{W}}$. The formulation in Eq A.5 is also a weighted variant of Hubbell's centrality [76]. Finally the formulation can be seen as an inverted weighted variant of the Page Rank algorithm [25]. Notice that this reduction to traditional network centrality formulations was only possible because of the way the Creativity Implication Network was constructed.

A.3.8 Originality vs. Influence

The formulation above sums up the two criteria of creativity, being original and being influential. We can modify the formulation to make it possible to give more emphasis to either of these two aspects when computing the creativity scores. For example it might be desirable to emphasize novel works even though they are not influential, or the other way around. Recall that the direction of the edges in Creativity Implication Network are no longer related to the temporal relation between the nodes. We can label (color) the edges in the network such that each outgoing edge $e(p_i, p_j)$ from a given node p_i is either labeled as a subsequent edge or a prior edge depending on the temporal relation between p_i and p_j. This can be achieved by defining two disjoint subsets of the edges in the networks

$$\begin{aligned} E^{\text{prior}} &= \{e(p_i, p_j) : t(p_j) < t(p_i)\} \\ E^{\text{subseq}} &= \{e(p_i, p_j) : t(p_j) \geq t(p_i)\} \end{aligned}$$

This results in two adjacency matrices, denoted by $\widetilde{W}^p$ and $\widetilde{W}^s$ such that $\widetilde{W} = \widetilde{W}^p + \widetilde{W}^s$, where the superscripts p and s denote the prior and subsequent edges respectively. Now Eq A.4 can be rewritten as

$$\begin{aligned} C(p_i) &= \frac{(1-\alpha)}{N} + \\ &\quad \alpha[\beta \sum_j \tilde{w}^p_{ij} \frac{C(p_j)}{N^p(p_j)} + (1-\beta) \sum_j \tilde{w}^s_{ij} \frac{C(p_j)}{N^s(p_j)}], \end{aligned} \tag{A.7}$$

where $N^p(p_j) = \sum_k \tilde{w}^p_{kj}$ and $N^s(p_j) = \sum_k \tilde{w}^s_{kj}$. The first summation collects the creativity scores stemming from prior nodes, i.e., encodes the originality part of the score, while the second summation collects creativity scores stemming from subsequent nodes, i.e, encodes influence. We introduced a parameter $0 \leq \beta \leq 1$ to control the effect of the two criteria on the result. The modified formulation above can be written as

$$C = \frac{(1-\alpha)}{N}\mathbf{1} + \alpha[\beta\widetilde{\widetilde{W^p}}C + (1-\beta)\widetilde{\widetilde{W^s}}C], \tag{A.8}$$

where $\widetilde{\widetilde{W^p}}$ and $\widetilde{\widetilde{W^s}}$ are the column stochastic adjacency matrices resulting from normalizing the columns of $\widetilde{W}^p$ and $\widetilde{W}^s$ respectively. It is obvious that the closed-form solution in Eq A.6 is applicable to this modified formulation where $\widetilde{\widetilde{W}}$ is defined as $\widetilde{\widetilde{W}} = \beta\widetilde{\widetilde{W^p}} + (1-\beta)\widetilde{\widetilde{W^s}}$.

A.3.9 Creativity Network for Art

In this section we explain how the framework can be realized for the particular case of visual art.

Visual Likelihood: For each painting we can use computer vision techniques to obtain different feature representations for its image, each encoding a specific visual aspect(s) related to the elements and principles of arts. We denote such features by f_i^a for painting p_i, where a denotes the visual aspect that the feature quantifies. We define the similarity between painting p_i and p_j, as the likelihood that painting p_j is coming from a probability model defined by painting p_i. In particular, we assume a Gaussian probability density model for painting p_i, i.e.,

$$S^a(p_j, p_i) = Pr(p_j|p_i, a) = \mathcal{N}(\cdot; f_i^a, \sigma^a I).$$

It is important to limit the connections coming to a given painting. By construction, any painting will be connected to all prior paintings in the graph. This makes the graph highly biased since modern paintings will have extensive incoming connections and early paintings will have extensive outgoing connections. Therefore we limit the incoming connections to any node to at most the top K edges (the K most similar prior paintings).

Temporal Prior: It might be desirable to add a temporal prior on the connections. If a painting in the nineteenth century resembles a painting from the fourteenth century, we shouldn't necessarily penalize that as low creativity. This is because certain styles are always reinventions of older styles, for example neoclassicism and renaissance. Therefore, these similarities between styles across distant time periods should not be considered as low creativity. Therefore, we can add a temporal prior to the likelihood as

$$S^a(p_j, p_i) = Pr^v(p_j|p_i, a) \cdot Pr^t(p_j|p_i),$$

where the second probability is a temporal likelihood (what is the likelihood that p_j is influenced p_i given their dates) and the first is the visual likelihood. There are different ways to define such a temporal likelihood. The simplest way is a temporal window function, i.e., $Pr^t(p_j|p_i) = 1$ if p_i is within K temporal neighbors prior to p_j and 0 otherwise[8].

[8] Alternatively, a Gaussian density can be use, $Pr^t(p_j|p_i) = \exp(-[t(p_i) - t(p_j)]^2/\sigma_t^2)$. However, adding such temporal Gaussian would complicate the algorithm since it will not be easy to estimate a suitable σ_t, specially the graph can have non-uniform density over the time line.

Balancing Function: There are different choices for the balancing function $B(w)$, as well as the parameter for that function. We mainly used a linear function for that purpose. The parameter m can be set globally over the whole graph, or locally for each time period. A global m can be set as the p-percentile of the weights of the graph, which is p-percentile of all the pairwise likelihoods. This directly means that p% of the edges of the graph will be reversed when constructing the Creativity Implication Graph. One disadvantage of a global balancing function is that different time periods have different distributions of weights. This suggests using a local-in-time balancing function. To achieve that we compute m_i for each node as p% of the weight distribution based on its temporal neighborhood.

A.3.10 Experiments and Results

A.3.11 Datasets and Visual Features

Artchive: This dataset was previously used for style classification and influence discovery [139]. It contains a total of 1710 images of art works (paintings and sculptures) by 66 artists, from 13 different styles ranging from AD 1412 to 1996, chosen from Mark Harden's Artchive database of fine-art [73]. The majority of these images are of the full work, while a few are details of the work.

Wikiart.org: We used the publicly available dataset of *"Wikiart paintings"*[9]; which, to the best of our knowledge, is the largest online public collection of artworks. This collection has images of 81,449 fine-art paintings and sculptures from 1,119 artist spanning from 1400 till after 2000. These paintings are from 27 different styles (Abstract, Byzantine, Baroque, etc.) and 45 different genres (Interior, Landscape, Portrait, etc.).

We pruned the dataset to 62,254 western paintings by removing genres and mediums that are not suitable for the analysis such as sculpture, graffiti, mosaic, installation, performance, photos, etc.

For both datasets the time annotation is mainly the year. Therefore, it is not possible to tell which is prior between any pair of paintings with the same year of creation. Therefore no edge is added between their corresponding nodes.

[9] http://www.wikiart.org/

We experimented with different state-of-the-art feature representations. In particular, the results shown here are using Classeme features [165]. These features were shown to outperform other state-of-the-art features for the task of style classification [139]. These features (2659 dimensions) provide semantic-level representation of images, by encoding the presence of a set of basic-level object categories (e.g. horse, cross, etc.), which captures the subject matter of the painting. Some of the low-level features used to learn the Classeme features also capture the composition of the scene. We also experimented with GIST features, which mainly encode scene decomposition along several perceptual dimensions (naturalness, openness, roughness, expansion, ruggedness) [116]. GIST features are widely used in the computer vision literature for scene classification.

A.3.12 Experiment Results

We show qualitative and quantitative experimental results of the framework applied to the aforementioned datasets. As mentioned in the introduction, any result has to be evaluated given the set of paintings available to the algorithm and the capabilities of the visual features used. Given that the visual features used are mainly capturing subject matter and scene composition, sensible creativity scores are expected to reflect these concept. A low creativity score does not mean that the work is not creative in general, it just means that the algorithm does not see it creative with respect to its encoding of subject matter and composition.

Figures A.22 shows the creativity scores computed for the Artchive dataset[10]. In this figure and all following figures we plot the scores vs. the year of the painting. The figures visualize some of the paintings that obtained high scores, as well as some with low scores (the scores in the plots are scaled). We randomly sampled points with low scores for visualization. A close look at the paintings that scored low (bottom of each plot) reveals the presence of typical subject matter, or in some cases the image presents an unclear view of a sculpture (e.g. Rodin 1889 sculpture in the bottom right of Figure A.22).

There are several interesting paintings that achieved high creativity scores. For example, the scream by Edvard Munch's (1893) scored very high relative to other paintings in that period

[10]For Figure A.22 a temporal prior was used. We set K=500, α=0.15.

(see Figure A.22). This painting is considered as the second iconic figure after Leonardo's Mona Lisa in the history of art, and it is known to be the most-reproduced painting in the twentieth century [82]. It is also one of the most outstanding expressionist paintings.

Figure A.24 shows a zoom-in plot to the period between 1850-1950, which is very dense in the graph of Figure A.22. We can see that Picasso's La Celestina (1903) scored the highest among his blue-period paintings. Picasso's Ladies of Avignon (1907) sticks out as high in creativity (obtained the highest score between 1904-1911). Art historians indicate that the flat picture plane and the application of primitivism in this painting made it an innovative work of art, which lead to Picasso's cubism [33]. We can notice a sharp increase in creativity scores at 1912, dominated by cubism work, with Picasso's Maquette for Guitar (1912) is the highest scoring in that surge. The up trend in creativity scores continues with several of Kasimir Malevich's first Suprematism paintings in 1915 topping the scores. This includes Malevich's Red square (1915), Airplane Flying (1915) and Black and Red square (1915) with almost identical scores at the top of this group, followed by Suprematist Construction (1915), Two-dimensional Self Portrait (1915), and Supermatist Composition (1915) - See Figure A.24 (the thumbnails of the last three paintings are not shown in the figure). Malevich's 1915's Black Square was not included in the analyzed collection that is used for this plot. However, his 1929's version of the Black Square was part of the collection and scored as high (see the blue star around the year 1929 in Figure A.24). The majority of the top-scoring paintings between 1916 and 1945 were by Piet Mondrian and Georgia O'Keeffee.

One of the interesting findings is the ability of our algorithm to point out wrong annotations in the dataset. For example, one of the highest scoring paintings around 1910 was a painting by Piet Mondrian called " Composition en blanc, rouge et jaune,"

(see the red-dotted-framed painting in Figure A.24). By examining this painting, we found that the correct date for it is around 1936 and it was mistakenly annotated in the Artchive dataset as 1910[11]. Modrain did not start to paint in this grid-based (Tableau) style untill around 1920. So it is no surprise that wrongly dating one of Mondrain's tableau paintings to 1910 caused it to obtain a high creativity score, even above the cubism paintings from that time. On

[11]The wrong annotation is in the Artchive CD obtained in 2010. The current online version of Artchive has corrected annotation for this painting

the Wikiart dataset, one of the highest-scored paintings was "Tornado" by contemporary artist Joe Goode, which was found to be mistakenly dated 1911 in Wikiart[12]. A closer look at the artist biography revealed that he was born in 1937 and this painting was created in 1991[13]. It is not surprising for a painting that was created in 1991 to score very high in creativity if it was wrongly dated to 1911. These two example, not only indicate that the algorithm works, but also show the potential of proposed algorithm in spotting wrong annotations in large datasets, which otherwise would require tremendous human effort.

Figure A.25 shows the creativity scores obtained for 62K paintings from the Wikiart datasets[14]. Similar to the figures above, we plot the scores vs. the year of the painting. We also randomly sampled points with low scores for visualization. The general trend in Figure A.25 shows peaks in creativity around late 15th to early 16th century (the time of High Renaissance), the late 19th and early 20th centuries, and a significant increase in the second half of the 20th century.

Originality vs. Influence - Analysis of Religious Paintings

In this experiment we investigate the effect of the two criteria of creativity: originality vs. influence. For this purpose we use the formulation in Eq A.8. In this experiment we used the religious paintings from the Wikiart dataset. This subset contains 5256 paintings in the period AD 1410-1993.

Figure A.26 & A.27 shows the creativity scores for this subset, where we set the parameter $\beta = 0.9$ to obtain the scores in Figure A.26 (i.e., emphasizing originality) vs. setting the parameter $\beta = 0.1$ to obtain the scores in Figure A.27 (i.e., emphasizing influence). From the figures we can notice that emphasizing originality biases the scores towards modern paintings, while emphasizing influence biases the scores towards earlier paintings in the collection. Comparing the same painting in the two figures can contrast its novelty vs. its influence. For example, Francisco Goya's Crucified Christ (1780) scored very high in Figure A.26, indicating its originality, and scored lower in Figure A.27 when measuring its influence. However, in both cases that painting gets higher creativity scores than other paintings from the same period.

---

[12] `http://www.wikiart.org/en/joe-goode/tornado-1911` - accessed on Feb 28th, 2015

[13] `http://www.artnet.com/artists/joe-goode/tornado-9-2Y7erPME95YlkhFp7DRWlA2`

[14] For Figure A.25 no temporal prior was used. We set K=500, α=0.15.

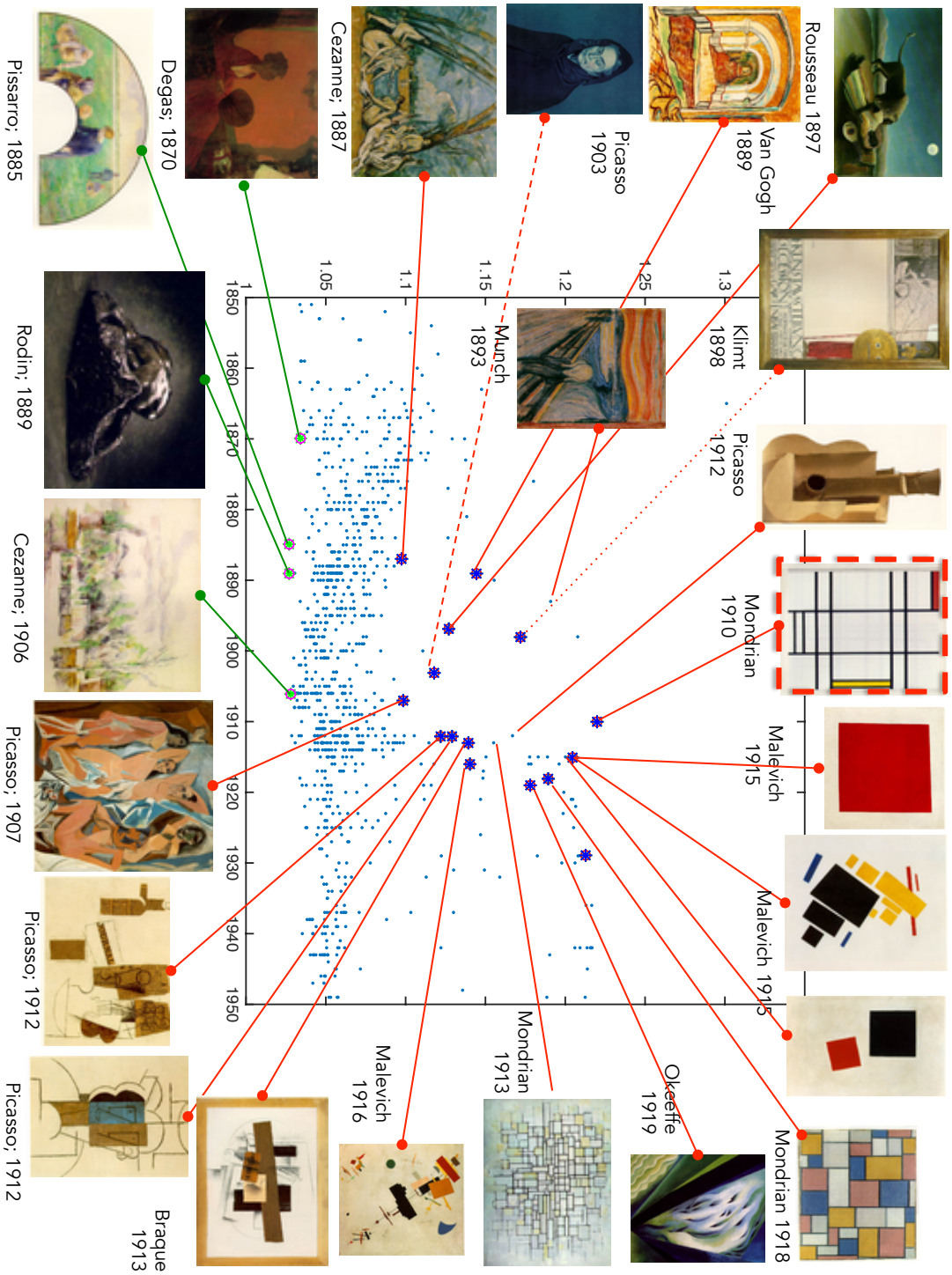

Figure A.24: Zoom in to the period of 1850-1950 from Figure A.22. Each point represents a painting. The horizontal axis is the year the painting was created and the vertical axis is the creativity score (scaled). Only artist names and dates of the paintings are shown on the graph because of limited space. The red-dotted-framed painting by Piet Mondrain scored very high because it was wrongly dated in the dataset to 1910 instead of 1936. See Section A.3.12 for a detailed explanation.

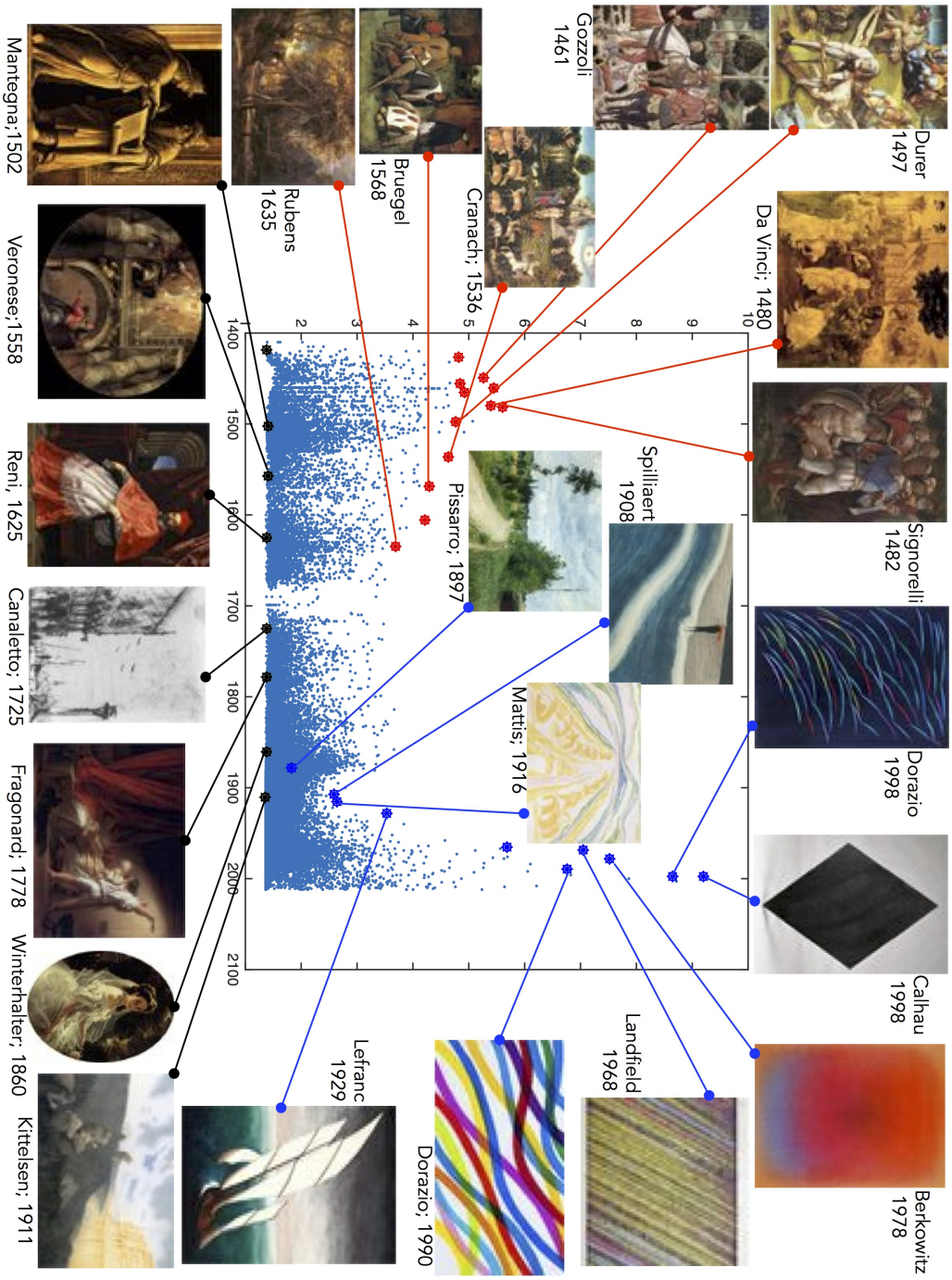

Figure A.25: Creativity scores for 62K painting from the Wikiart dataset. The horizontal axis is the year the painting was created and the vertical axis is the scaled creativity score.

It is clear that emphasizing originality results in an monotonically increasing upper envelop in the plot at the period from 1400 until around 1520 (see Figure A.27). This means that in this period there is a clear trend of increasing originality, where some paintings are pushing the upper envelop of the plot monotonically up. This up trend ends in the plot around 1520, which coincides with the end of the High Renaissance and the beginning of the Mannerism movement. An interesting example of the paintings in the up trend of originality between 1400-1520 is Andrea del Castagno's 1447 Last Supper[15] which is the earliest painting depicting the Last Supper in the analyzed collection (see Figure A.26). Domenico Ghirlandaio's 1476 last supper[16] scored higher along the upper envelop of the plot. In contrast, other versions of the Last Supper in the collection scored relatively lower, including Leonardo da Vinci's famous fresco. Out of 18 paintings by da Vinci in this collection his St. John the Baptist (1515) scored the highest (see Figure A.26). In the modern era, some of the paintings that scored very high in this religious collection are by Marc Chagall, Fernando Boetro, Salvador Dali and Nicholas Roerich (see Figures A.26 and A.27 for details).

Two-dimensional Creativity - Analysis of Portrait Paintings

Figure A.28 shows an example of two-dimensional analysis of creativity. In this experiment we used the subset of portrait paintings from the Wikiart dataset, which contains 12310 painting from the period AD 1420-2011. We analyzed creativity using the Classeme and GIST features as explained earlier, which yields two dimensions of creativity coordinates. Each point in the plot represents a single painting with two creativity scores. Unlike the previous figures, where we showed creativity vs. time, here we mainly show absolute creativity with respect to the two dimensions, i.e., we can not judge the relative creativity at any point of time from this plot. This makes the plot biased towards visualizing modern paintings. It is clear from the plot that the horizontal axis correlates with abstraction in the shape and form, while the the vertical axis correlates with texture and pattern.

---

[15] A fresco located at the church of Sant'Apollonia in Florence

[16] A Fresco located in the abbey of San Michele Arcangelo a Passignano in Tavernelle Val di Pesa, near Florence

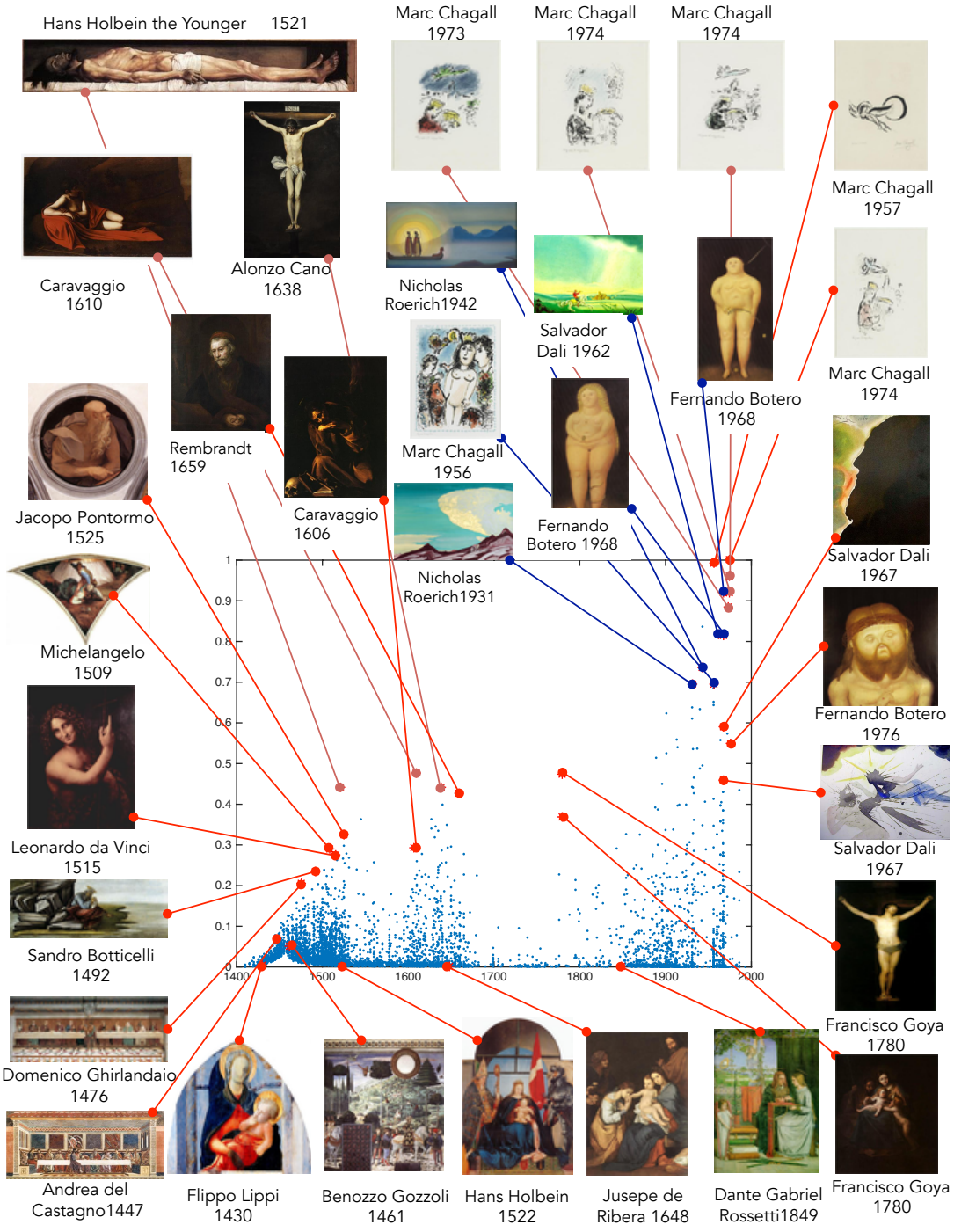

Figure A.26: Creativity scores for 5256 religious paintings from the Wikiart dataset (AD 1410-1993), emphasizing originality in computing the creativity sores. The horizontal axis is the year the painting was created and the vertical axis is the scaled creativity score.

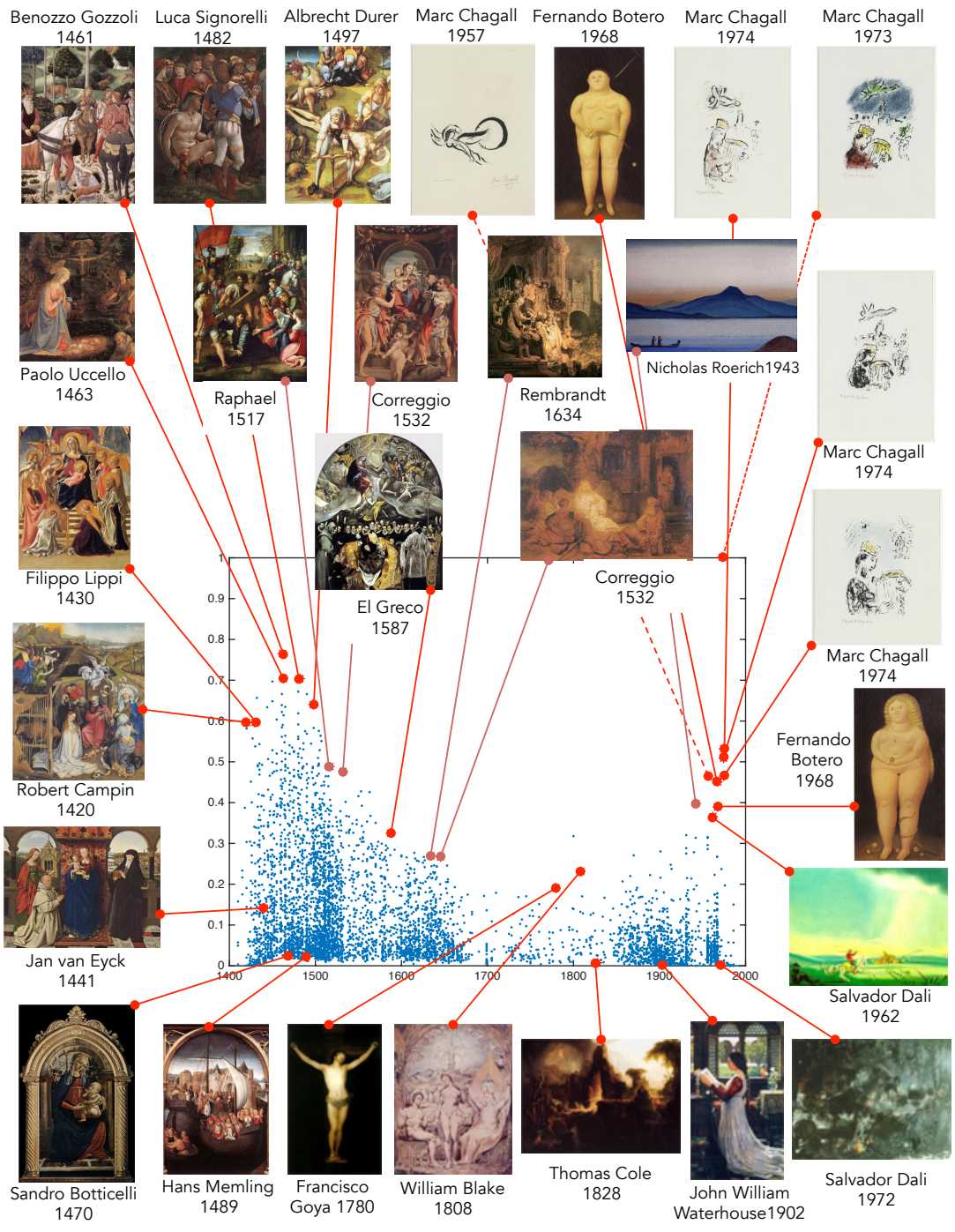

Figure A.27: Creativity scores for 5256 religious paintings from the Wikiart dataset (AD 1410-1993), emphasizing influence in computing the creativity sores. The horizontal axis is the year the painting was created and the vertical axis is the scaled creativity score.

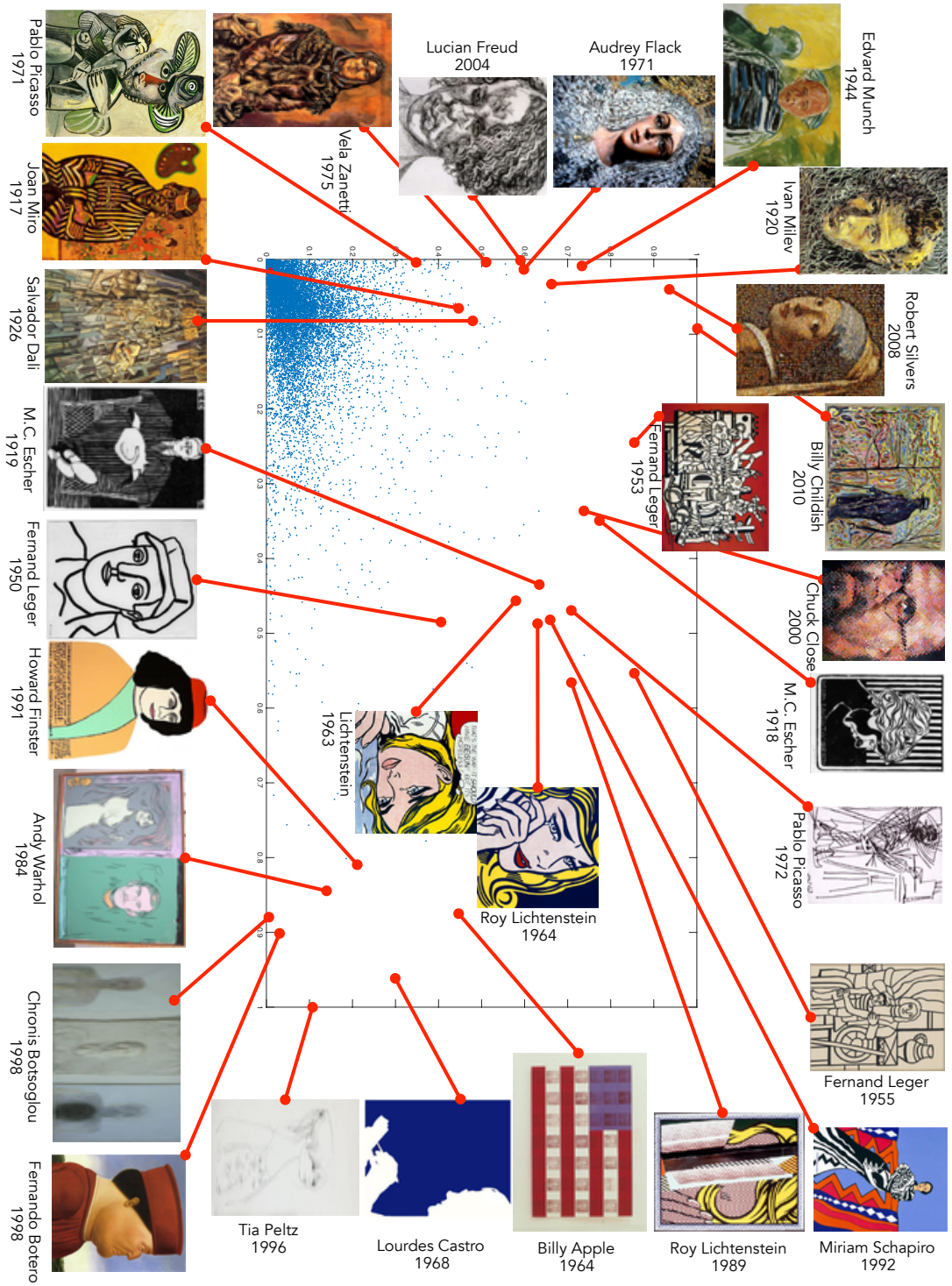

Figure A.28: Two dimensional creativity scores for 12310 portrait paintings from the Wikiart dataset, ranging from 1420 until 2011.

A.3.13 Time Machine Experiment

Table A.19: Time Machine Experiment

Art movement	avg % gain/loss	% increase
Moving backward to AD 1600		
Neoclassicism	5.78%±1.28	97%±4.8
Romanticism	7.52%± 2.04	98%± 4.2
Impressionism	14.66%± 2.78	99%±3.2
Post-Impressionism	16.82%±2.22	99%±3.1
Symbolism	15.2%±2.94	97%±4.8
Expressionism	16.83%±2.43	98%±4.2
Cubism	13.36%±2.43	89%±9.9
Surrealism	12.66%±1.82	95%±7.1
American Modernism	11.75%±2.99	84%±8.4
Wandering around to AD 1600		
Renaissance	0.68 %± 2.05	39%±5.7
Baroque	2.85%± 1.09	71%±19.7
Moving forward to AD 1900		
Renaissance	-8.13%± 2.02	20%±10.5
Baroque	-10.2%±2.03	0%±0

Given the absence of ground truth for measuring creativity and the aforementioned wrong time annotations inspired us with a methodology to quantitatively evaluate the framework. We designed what we call "time machine" experiment, where we change the date of an artwork to some point in the past or some point in the future, relative to its correct time of creation. Then we compute the creativity scores using the wrong date, by running the algorithm on the whole data. We then compute the gain (or loss) in the creativity score of that artwork compared to its score using correct dating. What should we expect from an algorithm that assigns creativity scores in a sensible way? Moving a creative painting back in history would increase its creativity score, while moving a painting forward would decrease its creativity. Therefore, we tested three settings: I) Moving back to AD 1600: For styles that date after 1750, we set the test paintings back to a random date around 1600 using Normal distribution with mean 1600 and standard deviation 50 years (i.e. $N(1600, 50^2)$). II) Moving forward to AD 1900: For the Renaissance and Baroque styles, we set the test paintings to random dates around 1900 sampled from $N(1900, 50^2)$. III) Wandering about AD 1600 (baseline): In this experiment, for the Renaissance and Baroque styles, we set the test paintings to random dates around 1600 sampled from $N(1600, 50^2)$.

Table A.19 shows the results of these experiments. We ran this experiment on the Artchive dataset with no temporal prior. In each run we randomly selected 10 test paintings of a given style and applied the corresponding move. We used 10 as a small percentage of the dataset (less than 1%), not to disturb the global distribution of creativity. We repeated each experiment 10 times and reported the mean and standard deviations of the runs. For each style we computed the average gain/loss of creativity scores by the time move. We also computed the percentage of the test paintings whose scores have increased. From the table we clearly see that paintings from Impressionist, Post-Impressionist, Expressionist, and Cubism movements have significant gain in their creativity scores when moved back to 1600. In contrast, Neoclassicism paintings have the least gain, which makes sense, because Neoclassicism can be considered as revival to Renaissance. Romanticism paintings also have a low gain when moved back to 1600, which is justified because of the connection between Romanticism and Gothicism and Medievalism. On the other hand, paintings from Renaissance and Baroque styles have loss in their scores when moved forward to 1900, while they did not change much in the wandering-around-1600 setting.

A.3.14 Conclusion and Discusion

The chapter presented a computational framework to assess creativity among a set of products. We showed that, by constructing a creativity implication network, the problem reduces to a traditional network centrality problem. We realized the framework for the domain of visual art, where we used computer vision to quantify similarity between artworks. We validated the approach qualitatively and quantitively on two large datasets.

The most important conclusion of this work is that, when introduced with a large collection of paintings (and sculptures), the algorithm can successfully highlight paintings that are considered creative (original and influential). The algorithm achieved that without any knowledge about art or art history encoded in its input. In most cases the results of the algorithm are pieces of art that art historians indeed highlight as innovative and influential. The algorithm achieved this assessment by visual analysis of paintings and considering their dates only.

Besides this qualitative evidence, we also proposed a methodology for validating the results of the algorithm through what we denote as time machine experiments. This experiments

quantitatively validated the proposed algorithm.

In this chapter we focused on “creative” as an attribute of a product, in particular artistic products such as painting, where creativity of a painting is defined as the level of its originality and influence. However, the computational framework can be applied to other forms such as sculpture, literature, science etc. Quantifying creativity as an attribute of a product facilitates quantifying the creativity of the person who made that product, as a function over the creator’s set of products. Hence, our proposed framework also serves as a way to quantify creativity as an attribute for people.

Clearly, it is not possible to judge creativity based on one specific aspect, e.g. use of color, perspective, subject matter, etc. For example it was the use of perspective that characterized the creativity at certain point of art history, however it is not the same aspect for other periods. This highly suggests the need to measure creativity along different dimensions separately where each dimension reflects certain visual aspects that quantify certain elements of art. The proposed framework can be used with multiple artistic concepts to achieve multi-dimensional creativity scoring.

References

[1] P. Agrawal, D. Stansbury, J. Malik, and J. L. Gallant. Pixels to voxels: Modeling visual representation in the human brain. *arXiv preprint arXiv:1407.5104*, 2014.

[2] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 1974.

[3] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *CVPR'13: IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[4] R. Arnheim. *Visual thinking*. Univ of California Press, 1969.

[5] R. S. Arora and A. M. Elgammal. Towards automated classification of fine-art painting style: A comparative study. In *ICPR*, 2012.

[6] F. G. Ashby and L. A. Alfonso-Reese. Categorization as probability density estimation. *Journal of mathematical psychology*, 39(2):216–233, 1995.

[7] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *ECCV*, 2012.

[8] Y. Bar, N. Levy, and L. Wolf. Classification of artistic styles using binarized features derived from a deep neural network. 2014.

[9] K. Barnard, P. Duygulu, and D. Forsyth. Clustering art. In *CVPR*, 2001.

[10] E. Bart and S. Ullman. Cross-generalization: Learning novel classes from a single example by feature replacement. In *CVPR*, 2005.

[11] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2002.

[12] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[13] A. Bentkowska-Kafel and J. Coddington. *Computer Vision and Image Analysis of Art: Proceedings of the SPIE Electronic Imaging Symposium, San Jose Convention Center, 18-22 January 2010*. PROCEEDINGS OF SPIE. 2010.

[14] I. E. Berezhnoy, E. O. Postma, and H. J. van den Herik. Automatic extraction of brushstroke orientation from paintings. *Machine Vision and Applications*, 20(1):1–9, 2009.

[15] A. Bergamo and L. Torresani. Classemes and other classifier-based features for efficient object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1, 2014.

[16] A. Bergamo, L. Torresani, and A. W. Fitzgibbon. Picodes: Learning a compact code for novel-category recognition. In *Advances in Neural Information Processing Systems*, pages 2088–2096, 2011.

[17] I. Bierderman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147, 1987.

[18] A. L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1):245–271, 1997.

[19] L. Bo, X. Ren, and D. Fox. Kernel descriptors for visual recognition. In *Advances in Neural Information Processing Systems*, pages 244–252, 2010.

[20] L. Bo and C. Sminchisescu. Twin gaussian processes for structured prediction. *IJCV*, 2010.

[21] M. A. Boden. *The creative mind: Myths and mechanisms*. Basic Books, 1990.

[22] I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2005.

[23] S. P. Borgatti and M. G. Everett. A graph-theoretic perspective on centrality. *Social networks*, 28(4):466–484, 2006.

[24] S. Branson, C. Wah, B. Babenko, F. Schroff, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *(ECCV)*, 2010.

[25] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1):107–117, 1998.

[26] R. S. Cabral, J. P. Costeira, F. De la Torre, A. Bernardino, and G. Carneiro. Time and order estimation of paintings based on visual features and expert priors. In *SPIE Electronic Imaging, Computer Vision and Image Analysis of Art II*, 2011.

[27] C. F. Cadieu, H. Hong, D. L. Yamins, N. Pinto, D. Ardila, E. A. Solomon, N. J. Majaj, and J. J. DiCarlo. Deep neural networks rival the representation of primate it cortex for core visual object recognition. *PLoS computational biology*, 2014.

[28] G. Carneiro. Graph-based methods for the automatic annotation and retrieval of art prints. In *ICMR*, 2011.

[29] G. Carneiro, N. P. da Silva, A. D. Bue, and J. P. Costeira. Artistic image classification: An analysis on the printart database. In *ECCV*, 2012.

[30] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman. Tricos: A tri-level class-discriminative co-segmentation method for image classification. In *European Conference on Computer Vision*, 2012.

[31] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*

[32] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[33] D. Cooper. *The cubist epoch*. Metropolitan Museum of Art, 1971.

[34] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *JMLR*, 2002.

[35] A. N. D. Blei and M. Jordan. Latent dirichlet allocation. In *Journalof Machine Learning Research*, 2003.

[36] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[37] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, 2007.

[38] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV*. 2010.

[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[40] J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[41] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *Neural Information Processing Systems*, 2013.

[42] K. Duan, D. Parikh, D. J. Crandall, and K. Grauman. Discovering localized attributes for fine-grained recognition. In *CVPR*, 2012.

[43] L. Duan, D. Xu, I. W.-H. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. *TPAMI*, 2012.

[44] M.-P. Dubuisson and A. K. Jain. A modified hausdorff distance for object matching. In *Pattern Recognition*, 1994.

[45] K. Ehinger, J. Xiao, A. Torralba, and A. Oliva. Estimating scene typicality from human ratings and image features. In *Proceedings of 33rd Annual Meeting of the Cognitive Science Society*, 2011.

[46] A. Elgammal and B. Saleh. Quantifying creativity in art networks. 2015.

[47] M. Elhoseiny, A. Elgammal, and B. Saleh. Write a classifier: Predicting visual classifiers from unstructured text descriptions. *arXiv preprint arXiv:1601.00025*, 2015.

[48] M. Elhoseiny, B. Saleh, and A. Elgammal. Write a classifier: Zero shot learning using purely textual descriptions. In *International Conference on Computer Vision (ICCV)*, 2013.

[49] I. Endres and D. Hoiem. Category-independent object proposals with diverse ranking. *PAMI*, 2014.

[50] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision (IJCV)*, 2010.

[51] M. V. Fahad Shahbaz Khan, Joost van de Weijer. Who painted this painting? 2010.

[52] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. 2010.

[53] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009.

[54] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *European Conference on Computer Vision*, pages 15–29. Springer Berlin Heidelberg, 2010.

[55] L. Fe-Fei, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *CVPR*, 2003.

[56] L. Fei-fei. A bayesian hierarchical model for learning natural scene categories. In *In CVPR*, 2005.

[57] J. Feldman. Bias toward regular form in mental shape spaces. *Journal of Experimental Psychology: Human Perception and Performance*, 2000.

[58] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2010.

[59] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba. Semantic label sharing for learning with many categories. In *ECCV*. 2010.

[60] L. Fichner-Rathus. *Foundations of Art and Design*. Clark Baxter.

[61] M. Fink. Object classification from a single example utilizing class relevance metrics. In *NIPS*, 2004.

[62] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[63] K. Fukushima. Artificial vision by multi-layered neural networks: Neocognitron and its advances. *Neural Networks*, 37:103–119, 2013.

[64] L. F. J. W. G. Csurka, C. Dance and C. Bray. Visual categorization with bags of keypoints. In *Proc. of ECCV International Workshop on Statistical Learning in Computer Vision*, 2004.

[65] M. Ghodrati, A. Farzmahdi, K. Rajaei, R. Ebrahimpour, and S.-M. Khaligh-Razavi. Feedforward object-vision models only tolerate small image variations compared to human. *Frontiers in computational neuroscience*, 2014.

[66] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[67] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *NIPS*, 2004.

[68] N. D. Goodman, J. B. Tenenbaum, J. Feldman, and T. L. Griffiths. A rational analysis of rule-based concept learning. *Cognitive Science*, 32(1):108–154, 2008.

[69] D. Graham, J. Friedenberg, and D. Rockmore. Mapping the similarity space of paintings: image statistics and visual perception. *Visual Cognition*, 2010.

[70] G. Griffin and P. Perona. Learning and using taxonomies for fast visual categorization. In *CVPR*, 2008.

[71] P. Guyer and A. W. Wood. *Critique of the Power of Judgement. The Cambridge Edition of the Works of Immanuel Kant*. Cambridge University Press, 2000.

[72] J. A. Hampton. Typicality, graded membership, and vagueness. In *Cognitive Science*, 2007.

[73] M. Harden. The artchive@http://artchive.com/cdrom.htm.

[74] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *arXiv preprint arXiv:1502.01852*, 2015.

[75] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 1970.

[76] C. H. Hubbell. An input-output approach to clique identification. *Sociometry*, pages 377–399, 1965.

[77] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

[78] W. L. I. Widjaja and F. Wu. Identifying painters from color profiles of skin patches in painting images. In *ICIP*, 2003.

[79] I. Jarvie. The rationality of creativity. In *Thinking about Society: Theory and Practice*, pages 282–301. Springer, 1986.

[80] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[81] C. R. Johnson, E. Hendriks, I. J. Berezhnoy, E. Brevdo, S. M. Hughes, I. Daubechies, J. Li, E. Postma, and J. Z. Wang. Image processing for artist identification. *Signal Processing Magazine, IEEE*, 25(4):37–48, 2008.

[82] P. Johnson. *Art: a new history*. Weidenfeld & Nicolson, 2003.

[83] A. A. E. A. Z. W. T. F. Josef Sivic, Bryan C. Russell. Discovering objects and their location in images. In *ICCV*, 2005.

[84] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *Computer Vision and Pattern Recognition(CVPR)*, 2013.

[85] Y. Keselman and S. Dickinson. Generic model abstraction from examples. *PAMI*, 2005.

[86] S.-M. Khaligh-Razavi and N. Kriegeskorte. Deep supervised, but not unsupervised, models may explain it cortical representation. 2014.

[87] A. Khosla, A. Raju S., A. Torralba, and A. Oliva. Understanding and predicting image memorability at a large scale. *International Conference on Computer Vision*, 2015.

[88] N. Krishnamoorthy, G. Malkarnenkar, R. Mooney, K. Saenko, U. Lowell, and S. Guadarrama. Generating natural-language video descriptions using text-mined knowledge. *NAACL HLT*, 2013.

[89] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[90] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011.

[91] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.

[92] S. Lad and D. Parikh. Interactively guiding semi-supervised clustering via attribute-based explanations. In *European Conference on Computer Vision (ECCV)*, 2014.

[93] B. M. Lake, W. Zaremba, R. Fergus, and T. M. Gureckis. Deep neural networks predict category typicality ratings for images. 2015.

[94] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by betweenclass attribute transfer. In *CVPR*, 2009.

[95] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 2015.

[96] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[97] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[98] J. Li and J. Z. Wang. Studying digital imagery of ancient paintings by mixtures of stochastic models. *Image Processing, IEEE Transactions on*, 13(3):340–353, 2004.

[99] J. Li, L. Yao, E. Hendriks, and J. Z. Wang. Rhythmic brushstrokes distinguish van gogh from his contemporaries: Findings via automated brushstroke extraction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012.

[100] T. E. Lombardi. The classification of style in fine-art painting. *ETD Collection for Pace University. Paper AAI3189084.*, 2005.

[101] S. Lyu, D. Rockmore, and H. Farid. A digital technique for art authentication. *Proceedings of the National Academy of Sciences of the United States of America*, 101(49):17006–17010, 2004.

[102] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[103] M. T. K. Michael W. Eysenck. *Cognitive psychology: a student's handbook*. Psychology press, 2005.

[104] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. In *CVPR*, 2000.

[105] G. A. Miller. Wordnet: A lexical database for english. *COMMUNICATIONS OF THE ACM*, 1995.

[106] J. P. Minda and J. D. Smith. Prototypes in category learning: the effects of category size, category structure, and stimulus complexity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27(3):775, 2001.

[107] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[108] G. L. Murphy. *The Big Book of Concepts*. MIT press, 2002.

[109] F. Murtagh and P. Legendre. Ward's hierarchical clustering method: Clustering criterion and agglomerative algorithm. *CoRR*, 2011.

[110] A. S. W. Myung Jin Choi, Antonio Torralba. Context models and out-of-context objects. In *To appear in Pattern Recognition Letters*, 2012.

[111] B. Nanay. An experiential account of creativity. In E. S. Paul and S. B. Kaufman, editors, *The Philosophy of Creativity: New Essays*. Oxford University Press, 2014.

[112] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes, 2001.

[113] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. 2015.

[114] M.-E. Nilsback and A. Zisserman. Automated flower classification over large number of classes. In *ICVGIP*, 2008.

[115] R. M. Nosofsky. Choice, similarity, and the context theory of classification. *Journal of Experimental Psychology: Learning, memory, and cognition*, 10(1):104, 1984.

[116] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 2001.

[117] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[118] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.

[119] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *CVPR*, 2011.

[120] D. Parikh and K. Grauman. Relative attributes. In *International Conference on Computer Vision*, 2011.

[121] S. N. Parizi, A. Vedaldi, A. Zisserman, and P. Felzenszwalb. Automatic discovery and optimization of parts for image classification. *ICLR*, 2015.

[122] S. Park, W. Kim, and K. M. Lee. Abnormal object detection by canonical scene-based contextual model. In *European Conference on Computer Vision (ECCV)*, 2012.

[123] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition(CVPR)*, 2012.

[124] E. S. Paul and S. B. Kaufman. Introducing the philosophy of creativity. In *The Philosophy of Creativity: New Essays*. Oxford University Press, 2014.

[125] N. Pinto, D. Cox, and J. J. DiCarlo. Why is real-world visual object recognition hard? 2008.

[126] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, 1999.

[127] G. Polatkan, S. Jafarpour, A. Brasoveanu, S. Hughes, and I. Daubechies. Detection of forgery in paintings using supervised learning. In *16th IEEE International Conference on Image Processing (ICIP)*, 2009.

[128] P. K. R. Sablatnig and E. Zolda. Hierarchical classification of paintings using face- and brush stroke models. In *ICPR*, 1998.

[129] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[130] L. J. Rips. Inductive judgments about natural categories. *Journal of verbal learning and verbal behavior*, 14:665–681, 1975.

[131] M. Rohrbach, M. Stark, G. Szarvas, and B. Schiele. Combining language sources and robust semantic relatedness for attribute-based knowledge transfer. In *Parts and Attributes Workshop at ECCV*, 2010.

[132] E. Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and categorization*. Lawrence Erlbaum, 1978.

[133] E. H. Rosch. Slow lettuce: categories, concepts, fuzzy sets, and logical deduction. *Concepts and Fuzzy Logic*, 2011.

[134] E. H. Rosch and C. Mervis. Family resemblances: studies in the internal structure of cateogies. *Cognitive Psychology*, 1975.

[135] R. Sablatnig, P. Kammerer, and E. Zolda. Structural analysis of paintings based on brush strokes. In *Proc. of SPIE Scientific Detection of Fakery in Art*. SPIE, 1998.

[136] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*. 2010.

[137] R. Salakhutdinov, A. Torralba, and J. B. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *CVPR*, 2011.

[138] B. Saleh. Wow! that looks strange: computational models for detection and reasoning about abnormalities in images. *AI Matters*, 2(3):16–17, 2016.

[139] B. Saleh, K. Abe, R. S. Arora, and A. Elgammal. Toward automated discovery of artistic influence. *Multimedia Tools and Applications*, 75(7):3565–3591, 2016.

[140] B. Saleh, K. Abe, and A. Elgammal. Knowledge discovery of artistic influences: A metric learning approach. In *International Conference on Computational Creativity (ICCC)*, 2014.

[141] B. Saleh, M. Dontcheva, A. Hertzmann, and Z. Liu. Learning style similarity for searching infographics. In *41st annual conference on Graphics Interface (GI) 2015*. Halifax, Nova Scotia, June 3-5, 2015, 2015.

[142] B. Saleh and A. Elgammal. A unified framework for painting classification. In *IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1254–1261. IEEE, 2015.

[143] B. Saleh, A. Elgammal, and J. Feldman. Incorporating prototype theory in convolutional neural networks. In *Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

[144] B. Saleh, A. Elgammal, and J. Feldman. The role of typicality in object classification: Improving the generalization capacity of convolutional neural networks. *arXiv preprint arXiv:1602.02865*, 2016.

[145] B. Saleh, A. Elgammal, J. Feldman, and A. Farhadi. Toward a taxonomy and computational models of abnormalities in images. In *The Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*, 2016.

[146] B. Saleh, A. Farhadi, and A. Elgammal. Object-centric anomaly detection by attribute-based reasoning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

[147] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *IPM*, 1988.

[148] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations (ICLR 2014)*, page 16. CBLS, 2013.

[149] A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson. A baseline for visual instance retrieval with deep convolutional networks. In *ICLR*, 2015.

[150] C. Shen, J. Kim, L. Wang, and A. van den Hengel. Positive semidefinite metric learning using boosting-like algorithms. *Journal of Machine Learning Research*, 13:1007–1036, 2012.

[151] F. Shi, X. Huang, and Y. Duan. Robust harris-laplace detector by scale multiplication. In *ISVC (1) Lecture Notes in Computer Science*.

[152] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. 2015.

[153] J. Sivic and A. Zisserman. Efficient visual search of videos cast as text retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*

[154] S. Sloman. Feature-based induction. *Cognitive Psychology*, 25:231–280, 1993.

[155] E. L. Spratt and A. Elgammal. Computational beauty: Aesthetic judgment at the intersection of art and science. In *ECCV 2014 Workshops, Part I, Proceedings of when vision meets art (VisArt) workshop, Lecture Notes on Computer Science number 8925*. Springer, 2014.

[156] D. G. Stork. Computer vision and computer graphics analysis of paintings and drawings: An introduction to the literature. In *Computer Analysis of Images and Patterns*, pages 9–24. Springer, 2009.

[157] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[158] C. W. Taylor. Various approaches to and definitions of creativity. *The nature of creativity*, pages 99–121, 1988.

[159] J. B. Tenenbaum, V. Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, 2000.

[160] R. Toldo, U. Castellani, and A. Fusiello. A bag of words approach for 3d object categorization. In *Proceedings of the 4th International Conference on Computer Vision/Computer Graphics CollaborationTechniques*, 2009.

[161] A. Torralba. *Contextual Influences on Saliency*, pages 586–593. Academic Press / Elsevier, 2005.

[162] A. Torralba and A. Efros. Unbiased look at dataset bias. In *CVPR*, 2011.

[163] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *PAMI*, 2008.

[164] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. 2003.

[165] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010.

[166] K. E. Van De Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1582–1596, 2010.

[167] A. Vedaldi and K. Lenc. Matconvnet – convolutional neural networks for matlab. *CoRR*, 2014.

[168] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. *arXiv:1411.4555*, 2014.

[169] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.

[170] K. Weinberger and G. Tesauro. Metric learning for kernel regression. In *Eleventh international conference on artificial intelligence and statistics*, pages 608–615, 2007.

[171] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 2009.

[172] K. Q. Weinberger, F. Sha, and L. K. Saul. Learning a kernel matrix for nonlinear dimensionality reduction. In *Proceedings of the twenty-first international conference on Machine learning*, page 106. ACM, 2004.

[173] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical report, California Institute of Technology, 2010.

[174] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, pages 3485–3492. IEEE, 2010.

[175] D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 2014.

[176] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval*, MIR '07, 2007.

[177] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *MULTIMEDIA*, 2007.

[178] S. Yang, L. Bo, J. Wang, and L. Shapiro. Unsupervised Template Learning for Fine-Grained Object Recognition. In *Advances in Neural Information Processing Systems*, 2012.

[179] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011.

[180] B. Yao*, A. Khosla*, and L. Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *(CVPR)*, 2011.

[181] D. Zeimpekis and E. Gallopoulos. Clsi: A flexible approximation scheme from clustered term-document matrices. In *In SDM*, 2005.

[182] N. Zhang, R. Farrell, and T. Darrell. Pose pooling kernels for sub-category recognition. In *CVPR*, 2012.

[183] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *ICCV*, 2013.

[184] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *arXiv:1412.6856*.

[185] B. Zhou, J. Xiao, A. Lapedriza, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Neural Information Processing Systems*, 2014.

[186] X. Zhu, C. Vondrick, C. C. Fowlkes, and D. Ramanan. Do we need more training data? *IJCV*, 2015.