COMPREHENSIVE PHOTOGRAPHIC COMPOSITION
ASSISTANCE THROUGH MEANINGFUL EXEMPLARS

A Dissertation in
Computer Science and Engineering
by
Farshid Farhat

© 2021 Farshid Farhat

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August 2021
The dissertation of Farshid Farhat was reviewed and approved by the following:

Jesse L. Barlow  
Professor of Computer Science and Engineering  
Dissertation Co-Advisor, Chair of Committee

James Z. Wang  
Professor of Information Sciences and Technology  
Dissertation Co-Advisor

Robert T. Collins  
Associate Professor of Computer Science and Engineering

Wang-Chien Lee  
Associate Professor of Computer Science and Engineering

Jia Li  
Professor of Statistics

Zihan Zhou  
Assistant Professor of Information Sciences and Technology

Chita R. Das  
Distinguished Professor of Computer Science and Engineering  
Department Head of Computer Science and Engineering
Many people are interested in taking good photos and sharing them with others. The size and value of visual content on the web are growing because people share their memories on social media and trade as non-fungible tokens with digital coins. Also, emerging high-tech hardware and software facilitate the ubiquitousness and functionality of digital photography and marketing. These trends lead to many challenging areas in visual content analysis, such as computational image aesthetics, composition-aware image retrieval, and meaningful feedback in photographic systems. Since people like to take better photos of themselves using an app in their handy device, there is a vast market demand for photographic composition assistance to evaluate the essential aspects affecting the beauty of a taken photo and convey meaningful feedback to users.

This dissertation investigates new scientific and applied computational photography methods for helping people interested in taking astonishing photos. Because composition matters in photography, researchers have leveraged standard composition techniques, such as the rule of thirds and the perspective-aware methods, in providing photo-taking assistance. To assess the aesthetic quality of photos computationally, researchers also attempted to manipulate the images to improve the aesthetic quality. However, composition techniques developed by professionals are far more diverse than well-documented methods can cover. Also, there is a lack of a holistic framework to capture important aspects of a given scene and help individuals by constructive clues to take a better shot in their adventure.

We leverage one of the aspects of image aesthetics in landscape photography which is a linear perspective, i.e., illustrating a 3D depth view as a 2D image. To analyze the linear perspective of a 2D image, we use a contour detector to recognize the vanishing lines, and then we cluster them to find potential vanishing points (VPs) accurately. Then, our proposed strength measure chooses the dominant VP among the potential VPs. We use this approach to provide on-site feedback to
users via an image retrieval system based on linear perspective. Also, we leverage the triangle technique widely used in photography. We manage a large portrait dataset for this study and retrieve triangle-shaped human poses from the dataset to help amateur photographers.

Finally, we leverage the underexplored photography ideas, which are virtually unlimited, diverse, and correlated. We propose a comprehensive fork-join framework, named CAPTAIN (Composition Assistance for Photo Taking), to guide a photographer with a variety of photography ideas. The framework consists of a few components: integrated object detection, photo genre classification, artistic pose clustering, personalized aesthetics-aware image retrieval, and style set matching. A large managed dataset crawled from a Website with ideas from photography enthusiasts and professionals backs CAPTAIN. The work proposes steps to decompose a given amateurish shot into composition ingredients and compose them to bring the photographer a list of related and valuable ideas that researchers have not explored in the past. The work addresses personal preferences for composition by presenting a user-specified preference list of photography ideas. The framework extracts ingredients of a given scene as a set of composition-related features ranging from low-level features such as color, pattern, and texture to high-level features such as pose, category, rating, gender, and object. Our composition model, indexed offline, provides visual ideas for the given scene, a novel model for an aesthetics-related recommender system. The matching algorithm recognizes the best shot among a sequence of photos concerning the user’s preferred style set. We have conducted many experiments on the proposed components and reported findings. Also, this study is backed by a comprehensive user study demonstrating that the work is helpful to those taking photos.
# Table of Contents

List of Figures ix

List of Tables xiii

Acknowledgments xiv

## Chapter 1
### Introduction
  1.1 Problem Definition .............................................. 3
  1.2 Sought Challenges .............................................. 4
    1.2.1 Photography and Subjectivity .............................. 5
    1.2.2 Abundance, Diversity, and Correlation of Ideas ......... 5
  1.3 The Approach .................................................. 6
  1.4 Applications .................................................. 7
  1.5 Summary of Contributions ..................................... 8
  1.6 Organization .................................................. 9

## Chapter 2
### Background and Related Work
  2.1 Background .................................................... 11
    2.1.1 Image Aesthetics in Psychology ........................... 11
    2.1.1.1 Gestalt Theory in Visual Perception ................. 12
    2.1.1.2 Gestalt Principals of Grouping ........................ 12
    2.1.1.3 Implied and Psychic Lines ............................ 14
    2.1.1.4 Color Constancy and Sensation ........................ 14
    2.1.2 Image Aesthetics in Art .................................. 15
    2.1.2.1 Rule of Thirds ........................................ 16
    2.1.2.2 Visual Balance ......................................... 17
    2.1.2.3 Lines ................................................ 18
### Chapter 3

**Detecting Vanishing Points in Natural Scenes**

3.1 Introduction ........................................... 29
3.2 Ground Truth Dataset ..................................... 32
3.3 Contour-Based Vanishing Point Detection ................. 33
   3.3.1 J-Linkage ........................................ 33
   3.3.2 Edge Detection via Contours ....................... 34
   3.3.3 Experiments ...................................... 36
      3.3.3.1 Comparison of Edge Detection Methods ...... 37
      3.3.3.2 Comparison with State-of-the-Art .......... 39
      3.3.3.3 Parameter Sensitivity ........................ 39
3.4 Selection of the Dominant Vanishing Point .............. 40
   3.4.1 The Strength Measure ............................. 40
   3.4.2 Experiments ...................................... 42
      3.4.2.1 Dominant Vanishing Point Selection ....... 42
      3.4.2.2 Dominant Vanishing Point Verification ... 42

### Chapter 4

**Toward Assisting with Portrait Composition**

4.1 Introduction ........................................... 44
4.2 Accelerating Dataset Processing in Distributed Systems . 45
   4.2.1 Scheduling Challenges ............................ 46
   4.2.2 Our Scenario .................................... 47
   4.2.3 Randomized Inter-Server Scheduling ............... 48
4.3 Intelligent Portrait Composition Assistance .............. 49
   4.3.1 Requirements .................................... 51
   4.3.2 Methodology ..................................... 53
   4.3.3 Implementation .................................. 53
      4.3.3.1 Low-to-High Level Features Extraction .... 53
4.3.3.2 Image Aesthetics Extensive Learning (iAEL) . . . . 57
4.3.3.3 Image Parametric Aesthetic Model (iPAM) . . . . 61
4.3.3.4 Image Aesthetics Improvement (iAI) ............... 62
4.3.4 Performance Accuracy ..................................... 64

Chapter 5
Comprehensive Composition
Assistance for Photo Taking ..................................... 66
5.1 Introduction ....................................................... 66
5.2 The Dataset ....................................................... 71
5.3 Photo Decomposition ......................................... 72
  5.3.1 Integrated Object Detectors (IOD) ....................... 73
    5.3.1.1 Value Unification ........................................ 74
    5.3.1.2 Hysteresis Fusion ........................................ 76
    5.3.1.3 Object Importance ....................................... 76
  5.3.2 Category Detector (CaDe) .................................. 77
    5.3.2.1 Top-down Hierarchical Clustering ................. 78
    5.3.2.2 Portrait Multi-class Categorization ........... 78
  5.3.3 Artistic Pose Clustering (ArPose) ....................... 79
    5.3.3.1 Pose Clustering .......................................... 80
  5.3.4 Construction of Composition Model ....................... 84
5.4 Composition of Visual Elements ................................. 85
  5.4.1 Similarity Scores and Normalization ................... 87
  5.4.2 User Preferences and Ranking ............................. 88
5.5 Matching .......................................................... 89
  5.5.1 Pose Shot .................................................... 89
  5.5.2 User Favorite Shot .......................................... 92
5.6 Experiments ....................................................... 93
  5.6.1 Dataset Properties ........................................... 93
  5.6.2 Decomposition Analysis ..................................... 94
    5.6.2.1 Object Detection ........................................ 94
    5.6.2.2 Pose Estimator ........................................... 95
    5.6.2.3 Scene Parser ............................................. 95
    5.6.2.4 Hysteresis Fusion ....................................... 96
    5.6.2.5 Portrait Category Detection ....................... 99
    5.6.2.6 Artistic Pose Clustering ............................... 99
  5.6.3 User Study for Composition .............................. 100
  5.6.4 Runtime Analysis ........................................... 103
Chapter 6
Conclusions and Future Work

6.1 Conclusions .................................................. 105
6.2 Future Directions ........................................... 106
   6.2.1 Genre Extendibility ................................. 106
   6.2.2 Enhancement in User Interaction .................. 106
   6.2.3 Clustering of Photography Ideas ................. 107
   6.2.4 Innovative Shot ................................. 107

Bibliography ................................................. 108
## List of Figures

<table>
<thead>
<tr>
<th>Figure No.</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>The borders of the garden cross at a point called vanishing point.</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>A woman crosses her ankles to make a triangle.</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>A guy with various poses.</td>
<td>4</td>
</tr>
<tr>
<td>1.4</td>
<td>While there are to-do’s and not-to-do’s for a simple shot, many ideas can create a nice pose. These ideas are diverse, and can have a correlation.</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>What can you see in the picture? Gestalt theory says brain can perceive a meaningful whole from a messy scene. [taken from: <a href="http://kut.org/post/gestalt-principles-and-why-we-search-whole">http://kut.org/post/gestalt-principles-and-why-we-search-whole</a>]</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Proximity Law of Grouping in Gestalt Theory [taken from Wikipedia].</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>Similarity Law of Grouping in Gestalt Theory [taken from Wikipedia].</td>
<td>13</td>
</tr>
<tr>
<td>2.4</td>
<td>Closure Law of Grouping in Gestalt Theory [taken from Wikipedia’s gestalt psychology page]</td>
<td>13</td>
</tr>
<tr>
<td>2.5</td>
<td>Implicit vanishing lines made of snow.</td>
<td>14</td>
</tr>
<tr>
<td>2.6</td>
<td>The Last Supper painting by Leonard da Vinci.</td>
<td>15</td>
</tr>
<tr>
<td>2.7</td>
<td>Rule of thirds states the subject should lie on a power line/point at 1/3 and 2/3 of each dimension.</td>
<td>17</td>
</tr>
<tr>
<td>2.8</td>
<td>Various types of balanced images: symmetric, radial [Flickr ID 6873558025] and asymmetric [Flickr ID 6555126819].</td>
<td>18</td>
</tr>
<tr>
<td>2.9</td>
<td>Diagonal lines intersect at subject faces [taken from Eric Kim photography website].</td>
<td>18</td>
</tr>
<tr>
<td>2.10</td>
<td>A girl creating the triangle technique by her hand gesture.</td>
<td>19</td>
</tr>
<tr>
<td>2.11</td>
<td>Simplicity technique emphasizes a single foreground object like a dish.</td>
<td>19</td>
</tr>
<tr>
<td>2.12</td>
<td>Leading lines vanishing at a point illustrates a feeling of a 3D view.</td>
<td>20</td>
</tr>
<tr>
<td>2.13</td>
<td>Framing technique using two fingers attracts viewer sight to the main subject.</td>
<td>20</td>
</tr>
<tr>
<td>2.14</td>
<td>Depth of field technique makes the foreground subject sharper than background.</td>
<td>20</td>
</tr>
</tbody>
</table>
3.1 Natural scene images with vanishing points. Images are from the “landscape” category of the AVA dataset. Manually labeled ground truth lines are marked in green.

3.2 Example natural scene images that are not suitable for this work. The first two images show diminishing perspective. The third image has two VPs. The last image contains parallel curves, not parallel lines.

3.3 Contour-based edge detection. (a) Original image. (b) The ultrametric contour map (UCM). (c) Filtered edges.

3.4 Vanishing point detection results. (a) Comparison to other edge detection methods and state-of-the-art detection methods. (b)-(d) Accuracy of our method w.r.t. parameters $\alpha$, $l_{\text{min}}$, and $\phi$, respectively.

3.5 Comparison of different edge detection methods. The four rows show the original images, and the edges detected by Canny detector, LSD, and our contour-based method, respectively. Yellow edges indicate the edges consistent with the ground truth dominant vanishing point.

3.6 Illustration of our edge strength measure.

3.7 Performance of the dominant VP selection method.

3.8 The impact of VP strength on the accuracy of dominant VP detection.

4.1 Portrait images given various scenes with several pose ideas for a better composed photo. Images from the 500px website are selected by our framework.

4.2 Schematic diagram of our proposed framework: The method consists of image aesthetics extensive learning (iAEL) to learn aesthetic reference features, image parametric modeling (iPAM) to capture image aesthetics dynamicity, and image aesthetic improvement (iAI) to optimizes and recommends the best feedback to photographer.

4.3 A portrait showing human body with different colors [taken from AVA dataset].


4.5 The results of CNN (1st row), KNN-SVM (2nd row), and our method (3rd row) for a sample shot at the left side.
5.1 Sample photos retrieved from the dataset based on the photo category and/or subject gender. Each retrieved result shows a collection of photography ideas that can be used by an amateur to compose photos for a given situation.

5.2 The flowchart of our composition assistance framework: Blue, black, red, and green flows show the user settings, the indexing process, the searching process, and the matching process, respectively. The decomposition box (the box with a dashed line) extracts the aesthetics-related features of the images and computes our composition model. The composition box (another box with a dashed line) recommends some well-composed images in our dataset based on the aesthetics-related features and user-specified preferences.

5.3 Qualitative results show the improvement by our integrated object detector (IOD). Each row includes the original, YOLOv3, OpenPose, PSPNet, and the results from our IOD.

5.4 Qualitative results of major clusters derived from our algorithm on the portrait dataset. Each row represents the top poses of each cluster.

5.5 Qualitative results of major clusters derived from the portrait dataset by DEC algorithm. Each row represents the top poses in each cluster. The DEC algorithm fails in clustering poses because the poses in the same cluster are not all consistent.

5.6 The dataset properties: (a) the logarithmic distribution of the view counts, (b) the distribution of the ratings, (c) the logarithmic distribution of the vote counts, and (d) the logarithmic distribution of the favorite counts.

5.7 The distributions of (a) the score obtained from the pose estimator, (b) the normalized area obtained from the pose estimator, and (c) the detection probability obtained from the object detector, and (d) the normalized area obtained from the object detector for our ground-truth images with a “person” as a common object by the pose estimator and the object detector.

5.8 For the ground-truth images without any “person”, the distributions of (a) the highest score (if any) obtained from the pose estimator, (b) the highest normalized area of the highest score object (if any) obtained from the pose estimator, and (c) the detection probability for the dominant object (if any) obtained from the object detector, and (d) the normalized area of the dominant object (if any) obtained from the object detector.

5.9 The 2D MAP surface w.r.t the normalized area of the pose estimator and the detection probability of the object detector as a heat map.
5.10 The result of the elbow method on the dataset. We could spot the elbow around 13-17 clusters. We are also showing the first derivative of the distortion, to show where it is going to flatten out.

5.11 Qualitative results of the composition step compared to [1]. Each row shows query image, results from [1], our method with equal weights, and ours with user-specified preferences (USP). The USP with respect to each row is: (1) \( W_{cade} = W_{gender} = 0.5 \), (2) \( W_{cade} = W_{vgg} = W_{iod} = 0.33 \), (3) \( W_{gender} = W_{vgg} = W_{cade} = 0.33 \), (4) \( W_{cade} = W_{iod} = 0.5 \), (5) \( W_{cade} = W_{gender} = W_{iod} = 0.33 \), (6) \( W_{cade} = W_{vgg} = W_{iod} = 0.33 \), (7) \( W_{cade} = W_{iod} = 0.5 \) and (8) \( W_{cade} = W_{vgg} = W_{iod} = 0.33 \).

5.12 A subset of image queries used for user study based on different types of categories such as semantics, single vs group, full-body, upper-body, facial, male vs female, etc.

5.13 The histogram of the acceptability rate based on the recommended photos versus the total number of recommendations (recommendation probability) compares ours with other methods.
List of Tables

5.1 The accuracy comparison between our object detector model versus the YOLOv3 and Faster R-CNN on our dataset to detect some known objects. ......................................................... 95
5.2 The comparison between our pose estimator versus OpenPose and HRNet on our dataset to detect body parts. ................................. 95
5.3 The accuracy comparison between our scene parser versus PSPNet and DeepLab with 101-depth ResNet on our dataset. ................. 96
5.4 The accuracy results of our category detector (CaDe) for ground-truth images compared to a 16-feature-based baseline. ............... 99
5.5 The summary of the methods used in CAPTAIN with respect to extraction algorithm, effect on retrieval, and improvement over baseline. .......................................................... 100
Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor Professor James Z. Wang, for his continuous support of my PhD study and research and his inspiration and motivation. He has been a distinguished advisor and a true friend. I would never have done my PhD at Penn State without his support.

My sincere thanks also go to my co-advisor, Prof. Jesse Barlow, who has provided me with valuable support. Besides my advisors, I would like to thank the members of my dissertation committee, Dr. Jia Li, Dr. Zihan Zhou, Dr. Robert Collins, and Dr. Wang-Chien Lee, for their time reviewing my dissertation and comments.

I would also like to thank my family for their support throughout my entire life. In particular, I must acknowledge my wife and best friend, Diman Zad Tootaghaj, without whose love, encouragement, and editing assistance I would not have finished this thesis. Furthermore, I wish to express my deepest gratitude to my family, Ashraf Hoseini and Mohammad Farhat, and my sister, Mahshid Farhat, for their endless love and support throughout my life. Furthermore, I would like to thank my parents-in-law, Soraya Pirani and Ebrahim Zad Tootaghaj, for their constant kindness and support. Finally, I am also grateful to my brothers-in-law, Saman and Siamand.

I also would like to thank the wonderful friends who became my second family in State College. My special thanks go to Sanam Sanei and Amirali Kanî, Sahar Zarmehri and Farid Farrokhi, Nima Zolghadr, Ardalan Mirshani, Farid Tayari, Neda Nasiriani and Alireza Haghghat, Roozbeh Kermani and Farina Fooladi, Narges Shahidi and Amir Meymand, Azadeh Keivani and Ashkan Balouchi, Shahrzad Fadaei, Ashkan Jasour, Laleh Sarkarat and Amineh Aref, Hana Khamfroush and Sam Heshmati, Amin Jadidi, Mohammad Kamani and Artemiz Jahanara, Athar Sefid and Hamid Firooz, Farnaz Tehran and Mehdi Kiani. My gratitude also extends to my colleagues, Baris Kandemir, Yukun Chen, Xinye Zheng, Zhuomin Zhang, and Luo Yu, for inspiring discussions.
I visited the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in the summer of 2016, and I offer my sincere thanks to Dr. Ruqian Lu and his research group for their support.

My PhD study and my visit to China were funded by the National Science Foundation grant number OAC-1027854. Also School of Electrical Engineering and Computer Science, The Penn State Fund for Innovation, and Simplr Technologies LLC funded my PhD study. Computation of this work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1548562. My achievements do not show the source of the funding agencies. Nevertheless, I am grateful for their support during my study.
Dedication

This dissertation is dedicated to my spouse (Diman), my daughter (Tara), and my family (Ashraf, Mohammad, and Mahshid) for their endless love, support, and encouragement.
Human knowledge is a tiny subset of the whole information domain, and it is familiar and meaningful to most humans, as they learn it during their life. Patterns can define the first notion of knowledge, e.g., a lion in a messy jungle. Patterns are more crucial than basic information because an escort robot may warn that a lion is coming after recognizing a pattern. The exploited knowledge from the patterns is its application, such as commercial benefits, safety, and aesthetics.

Detection and recognition techniques give us insights that computer vision and pattern recognition are two non-separable parts. The essence of computer vision is the use of media (including image, video, or voice) to detect patterns to extract information. Thus, it is not only about collecting data from the surrounding environment, but also it is about how to process and refine this data to exploit practical knowledge.

Nowadays, the amount of media shared on the Internet with the help of photo/video sharing companies increases rapidly each year. This trend causes a transition for big data from terabytes to petabytes and even exabytes. On the other hand, computational power is growing because of more advanced computer architectures (like GPU). As a result, we cannot only perform text-processing, but we can also analyze the visual data. Such capabilities, coupled with the availability of big visual data, lead to new algorithms for detecting, analyzing, and retrieving desirable patterns.

This dissertation focuses on image aesthetics assessment to help users take better
photographs through some meaningful exemplar recommendations. One of the aesthetic features in still images is perspective. For example, a photo taken from nature is a 2D projection of a 3D scene, so the projected parallel objects from the surrounding environment may cross each other in the affected dimension to show the depth of the objects. In reality, they are not always straight lines, and the source/sink of the objects may be out of the frame. For example, in Figure 1.1, there is no explicit vanishing line, but the color contrast between the flowers and the road creates two borders for the garden, which are vanishing at a point called the vanishing point. The detection of the vanishing points in natural scenes is very challenging, and it is explained in more detail in Chapter 3.

The work in Chapter 4 focuses on aesthetic features in portrait images. A portrait not only contains a face but also may contain the human body, including the head, trunk, arms, hands, and feet. The beauty of a portrait depends on the foreground positions of the human parts and the constellation of the background objects. For example, Figure 1.2 shows a woman standing with crossed ankles. This pose creates
a nice triangle making the photo more appealing. Specifically, we want to address the aesthetic evaluation of the human poses in portrait photography and improve the next shot’s quality by providing meaningful and constructive exemplars for a photographer.

The last work is a comprehensive approach to give exemplar recommendations in a broader range of aesthetic qualities. Applications include making better landscape and portrait photographs. The proposed framework detects the first shot ingredients and brings exemplar photographs based on these ingredients and the user’s preferences in photography.

Now, we define the problem and explain the intuition and the challenges that motivate us to come up with our approach to solve the problem. In the following sections, we describe our practical method that intelligently helps people capture beautiful photos from a scene in a reasonable amount of time.

1.1 Problem Definition

To capture an aesthetically appealing photo, one needs to compose many visual elements. For instance, the beauty of a portrait depends on the foreground positions of the human limbs and the constellation of the background objects. As illustrated in Figure 1.3, the photographer tried to create three shots by composing different
visual elements, including the rule of space, color palette, and a ballerina posture.

Our composition assistance aims to aid people in capturing a better shot given his or her current setup of the photography location. The input data is an amateurishly taken photo by a photographer or an automatically captured photo from the camera viewfinder. The output data is informative feedback (e.g. image, animation, and comment) that guides the photographer to take better shots and select the best match. Helpful feedback as a piece of side information can be any highly-rated photo taken in a pretty similar situation while having a better composition for the location/setup. We believe that there is at least one photography idea behind each feedback because the creator of the photo as a good photographer usually has his/her opinion(s), and each of their photos may contain a collection of new ideas. Many of them are made online for sale.

1.2 Sought Challenges

People have various photography tastes shaped since childhood, and the number of ideas for taking photos is not limited to an actual number. In this work, we encounter the following challenges that make the problem more challenging and more interesting.
1.2.1 Photography and Subjectivity

The challenging part of the problem is that feedback to an amateur photographer is not the only solution. There is also a feasible region of the solution because photography is subjective. Expressly, there is no unique photography idea for scenery, and based on various unseen tastes of the subject, there may be a range of related ideas. As shown in Figure 1.3, for a given location, these are just three shots by one photographer, but people may like all, some, or none of them!

Most of image aesthetics studies focus on image assessment and manipulation of captured photos based on the common techniques as mentioned in Chapter 2, but the important point is that the quality of photography cannot be quantified by one number, as people have different perception of the same taken photo. As a result, the active helper for an amateur photographer should be not only innovative but also considerate about personal preferences.

To cover personal preferences, available photographic feedback systems [2–6] have limitations in embracing a broad range of photography ideas and filtering unwanted categories. For instance, a general retrieval system [2–4] consists of mixed photography categories, including portrait, landscape, closeup, to name but a handful. Hence, this leads to an unrelated output from the input or feedback, which is limited to a narrow range of topics such as photos with vanishing points [5] or photos having triangles [6]. So, the currently available frameworks cannot remedy the beginners’ thirst for getting preferred professional-looking snapshots.

1.2.2 Abundance, Diversity, and Correlation of Ideas

Inspired by the standard strategy in professional photography [7–9] — artists gradually make a subject perfect for the last shot while they usually have a “to-do” list and a “not-to-do” list in their mind. As shown in Figure 1.4, each pose has many details. However, the difference is that we do not assume our amateur photographer has access to a studio to compose a new environment for the subject. The background is static or naturally composed before. For example, when we are in the woods, the trees and sky are invariant for our abilities. However, human bodies, animals, or some objects are posable, dynamic, or movable.

Furthermore, the number of photography ideas for any given location is not
limited to one shot, and they are also correlated across various locations. Even if we assume that the number of photography ideas in the world is limited, this number would be very high (e.g. over 10K) and emerging every day. To our knowledge, the performance of the deep learning models to classify an idea among a high number of correlated ideas degrades substantially. Similarly, there is no accurate food category detector from dish images because the number of food categories is high (e.g. over 65K) [10]. So, retrieving related photography ideas based on an input scenery is almost computationally impossible at the moment.

1.3 The Approach

To achieve our goal, we suggest the decomposition of the input scenery into as many essential ingredients. Like a chess puzzle, we should understand the constellation of the scene and then move toward the best position. Similarly, we decompose the input shot from the amateur photographer or the camera viewfinder into many visual elements, including low-level features such as color, texture, and high-level features such as semantic objects, photo style, and human pose with photo properties like tags and rating.

Then, we introduce the composition of the relaxed space of the extracted visual elements, and we retrieve related photography ideas from the photo dataset based on user-specified preferences (USP). To address the subjectivity of the photography
tastes, we introduce USP that personalizes the retrieval output to give a higher rank (lower cost) to the user favorite results.

After finding the photography ideas based on the current scene ingredients, in the next step of our method, called style set matching, we match the subject via viewfinder with the available ideas and user-selected style set. Then, we automatically shot the scene, similar to the “smile shot” mode in smart cameras.

Our method is not only limited to any specific photography rule but also generalized to any innovative masterpiece based on the ingredients of the taken photo. In our opinion, it has the potential to be used for other photography genres such as architectural and closeup. The flow of the proposed framework (Figure 5.2) includes the dataset indexing to accelerate the retrieval process, the image searching to retrieve photography ideas, and the composition matching to catch the best shot.

1.4 Applications

The ideas of this work stem from the civil applications of the work. One can find the usage of the aesthetic features such as artistic pose and perspective in old paintings like in Pietro Perugino’s work, The Delivery of the Keys. Also, recently these are very popular in photography, camerawork, artistic affairs, and robotics. Our visionary goal is to help an amateur photographer to take a better shot using our on-site feedback system for his/her former shot.

Specially vanishing point and lines detection are used in many applications such as scene reconstruction [11, 12], plane rectification [13], pose estimation [14, 15], camera calibration [16, 17], and autonomous navigation [18]. While perspective detection and exploitation have been extensively studied and surveyed [19], aesthetic aspects of natural images is still new and progressing fast. Nevertheless, this way, there are some shortages, as the related work does not consider challenging natural images. As a result, the following points are our contributions in VP detection work:

- Developing a new strength metric for VP detection.
• Assisting the photographer with our new exemplar feedback via viewpoint-specific image retrieval.
• Newly labeled dataset with vanishing lines for future research.

Another aspect of image aesthetics that we investigate is the photographic detection of pose composition. Human pose estimation is an old classic problem in computer vision, which is deeply investigated. Nowadays, since it is frequently used in human-computer interaction (HCI), gadget-free gaming devices like Kinect, virtual reality devices like HoloLens, security, and healthcare. There are many studies trying to improve the performance accuracy in single depth images [22–24] as well as 2D still images [25–27]. From an image aesthetics perspective, our newly proposed work focusing on portrait photography to improve human pose using on-site feedback for an amateur photographer. We want to manage the following contributions in this work:

• Investigating new aesthetic features related to portrait photography.
• Learning from extrapolated aesthetic features.
• Image parametric aesthetic model capturing the dynamicity.
• Improve the quality of the image using pose changing.
• Qualitative evaluation of a parametric aesthetic model.

1.5 Summary of Contributions

The innovation of this dissertation starts from the detection of the dominant vanishing point in natural landscape images using our proposed strength measure. We have developed a new vanishing point detection method containing an improved contour-based edge detector with J-Linkage. We show that it outperforms the state-of-the-art methods on our annotated dataset. The detected dominant vanishing points and the underlying vanishing lines leverage some information about a standard principle in photography called the leading lines, which help us with photo composition. As a practical application, new perspective-aware image retrieval has derived from our landscape dataset, which gives meaningful on-site feedback to an amateur photographer.
In another line of this dissertation, we have collected a large dataset for portrait photography ideas. We introduce a new framework for portrait composition assistance that aids amateur photographers in capturing a better shot. As the number of photography ideas is increasingly high, directly retrieving and matching the viewfinder photo with an image in our dataset is not straightforward. Furthermore, the retrieving system finds similar images and searches for images with similar semantics through decomposition and composition stages. After providing helpful feedback for a photographer, the camera matches the final pose with one of the exemplars and makes a portrait shot. Our user study has evaluated the performance of our framework. Also, the current portrait dataset has been used in another work to retrieve triangular portraits. The work results assist people interested in portrait photography with the triangle technique to receive better-posed triangular portraits as useful on-site feedback.

We have extended our dataset to cover landscape photography ideas as well. Our ultimate framework for comprehensive composition assistance guides people through exemplar photos for taking better shots. We have exploited and integrated many improved object detectors, sub-genre categorization, and pose clustering to extract useful aesthetics-related information. The resulting composition model can give better-recommended images because the output depends on more informative decomposed data from query images and user-specified preferences. The matching algorithm finds the best shot among a sequence of shots taken by the camera, which is more similar to the preferred style set selected by the user. Qualitative results give enough intuition about the functionality of the method, and experimental results demonstrate the framework’s performance.

1.6 Organization

In this chapter, we briefly introduced our stimulus to propose this approach, and then we described our goals and achievements in this way. Later in this proposal, we cover the following aspects in the next chapters:

- Introduction in this chapter (Chapter 1).
- Background and related work in Chapter 2.
• Detecting vanishing points in natural scenes with application in photo composition in Chapter 3.

• Portrait composition assistance works in Chapter 4.

• Comprehensive composition assistance for photo taking in Chapter 5.

• Conclusions and future works in Chapter 6.
Chapter 2 | Background and Related Work

This chapter explains the background behind visual aesthetics and composition in different fields, including art, psychology, and computer science. Then, the most recent related work is described.

2.1 Background

Image aesthetics is about the study of artwork exploited to take a photo. Then the dominant factors making the photo a masterpiece are investigated, and finally, we can learn the philosophy of the beauties used in work. One may figure out the beauty of work by emotion and intuition, but it is not common among all ordinary people, i.e., it is highly subjective. Therefore, using the assessment procedure of image aesthetics ease the scientific way to recognize the amount of art residing in work.

2.1.1 Image Aesthetics in Psychology

What connects image aesthetics and psychology is visual perception [28]. One of the photographers’ goals is to convey a visual concept to the viewer. On the other hand, the viewer wants to erase any randomness from the viewpoint and relates the visible objects. Usually, the designer provides some signs to guide the viewer; otherwise, the artwork would be boring.
2.1.1.1 Gestalt Theory in Visual Perception

Visual perception [29] tries to investigate how the brain can perceive knowledge about what eyes see. Psychologists have done many empirical studies in this area. Still, the notable finding in visual perception might be Gestalt psychology [30], i.e., what abilities of the brain make it to discover a meaningful whole from a chaotic environment. Figure 2.1 shows an example photo where you can find many objects when you concentrate more closely.

Figure 2.1: What can you see in the picture? Gestalt theory says brain can perceive a meaningful whole from a messy scene. [taken from: http://kut.org/post/gestalt-principles-and-why-we-search-whole]

2.1.1.2 Gestalt Principals of Grouping

Gestalt psychology says that grouping principles or laws can capture the ability of the brain to perceive parts as a whole. Some of the known grouping laws are as follows:

**Proximity:** The proximity law says that the brain can understand an assortment of the parts because of their proximity [31, 32]. It can be seen in Figure 2.2, the close circles form three identical groups of 12 and a group of 36.

**Similarity:** The similarity law says that the brain can group similar subsets of parts and see them as a whole [32]. For example, from Figure 2.3, you can discover three groups of black circles and three groups of white circles because of their color similarity.
Closure: The closure law says that the brain can recover missing parts of a whole and construct it to perceive it as a whole again [31–33]. This ability can give a big point to the brain rather than an artificially intelligent system. For example, Figure 2.4 shows an incomplete circle and rectangle, which anyone quickly detects.

There are other laws of grouping that we list and refer you to the references if you want to follow them carefully: symmetry, common fate, continuity, good gestalt, and past experience [31–33].
2.1.1.3 Implied and Psychic Lines

Lines are ubiquitous in human-made designs. From a psychological perspective, an implied line is a series of points establishing a line seen by the eyes and perceived by the brain. For example, Figure 2.5 illustrates a road getting narrower because its illusionary sides made of snow vanish towards a tree.

![Implicit vanishing lines made of snow.](image)

Figure 2.5: Implicit vanishing lines made of snow.

The lines in reality sometimes are hidden or not explicit. The imaginary ability of the brain will connect the objects intuitively. A psychic line is a line that does not exist, but you feel a rationale relation among the parts. As you can realize in Figure 2.6, The Last Supper by Leonardo da Vinci, there are many psychic lines between the apostles standing around Jesus, as they are gazing or pointing at each other.

2.1.1.4 Color Constancy and Sensation

We know from optical physics that the color of an object is created by photons radiating/reflecting from the object, so it is very dependent on the Sun’s position
because most of these photons are made from it. However, the brain tries to perceive the color by the kind of object, i.e., the brain usually thinks that water is transparent, a leaf is green, or the sky is blue. However, the sea is getting blue color from the sky, and the leaf is more yellow at noon, and the sky is dark at night. This kind of psychological feeling about the color of the objects is called color constancy.

Another psychological dimension of color is the temperature feeling. Each color may emphasize something in your body because your brain has it as an experience before. For example, if you live in a red-colored (blue-colored) office, you may feel hot (cold). On the other hand, most people feel happy when they see bright colors and feel sad when they surround by dark colors.

### 2.1.2 Image Aesthetics in Art

The promising way to evaluate image aesthetics is to trust giant artists in the field. We can rely on artistic works and try to find the hidden art inside their works. Nowadays, many conventional techniques to compose an artwork as design basics are identified in the literature [34, 35]. Some of these design basics are as follows:

- **Unity**: The rule of unity is nearly equivalent to a subset of grouping laws in gestalt theory. This rule states that there should be an organization or wholeness across the design parts.

- **Focal point**: Having a focal point in design attracts the attention of the viewer more closely and emphasizes the focal point. It is also widespread in
close-up photography.

- **Scale:** Using a reference object (like a wall clock) with other objects in a design will give an intuition about the scale of the objects compared to the reference, and we can also have a sense of a three-dimensional design.

- **Balance:** The rule of balance is a generalization of symmetry in a design. We do not search for exact symmetry, but we can find an equal amount of stuff with some sense of similarity in two sides of a design, which is a notion of balance.

- **Rhythm:** Design principle related to repetition is rhythm, where the designer tries to convey his/her intention to the viewer by replicating a particular pattern in his/her design.

- **Other elements of a design:** The other valuable elements in the design are just listed as line, shape, pattern and texture, an illusion of space and motion, value, and color.

A subset of these rules can be used in image aesthetics and photography as an art of composition [36, 37], which we are interested in explaining as follows because they are more related to our focus in the proposed work.

### 2.1.2.1 Rule of Thirds

Rule of thirds is the most famous photography rule from the spatial composition aspect. This rule states that the most prominent part of the picture should reside on or close to one of the four power lines or four power points. These two pairs of power lines divide the picture into nine equal-sized rectangles by two parallel horizontal lines and two parallel vertical lines. The four power points are the intersections of these two pairs of parallel lines.

The supporters of the rule of thirds believe that residing the photo’s subject on these lines or closely aligning with them makes the work more appealing than putting the subject at the center. This rule can be categorized as a spatial composition angle, and perhaps it has stemmed from the golden ratio as it is very close to the $3/2$ value. It wants to fit the main subject with the weight of $1/3$ in an optimal position while having the other parts in the background with the weight of $2/3$. 


Figure 2.7: Rule of thirds states the subject should lie on a power line/point at 1/3 and 2/3 of each dimension.

Figure 2.7 shows the realization of this rule where the butterfly as the foreground is not positioned at the center; instead, it is mostly on one of the power lines and two power points.

2.1.2.2 Visual Balance

The other important aesthetic rule in advanced photography is visual balance. The definition of visual balance in an artwork is not as easy as the rule of thirds because there are no accurate borders for the definition. The trivial version of visual balance is symmetrical balance when one side of the image is the reflection of the other side with respect to the reflection line. If some patterns in an image can be symmetrically transformed to each other with respect to a point, it is called radial balance. However, the challenging kind of visual balance is an asymmetric balance when the most salient object seems closer than the other salient object. Hence, the most salient is bigger in a 2D projected image and closer to the center, but in reality, they are more alike from another angle of view [34].

Figure 2.8 shows sample images with mentioned types of visual balances, respectively, including symmetrical, radial, and asymmetrical balances. To find the balanced point, we can calculate the center of the mass of the saliency map of the image. Similarly, the balanced line can be the regression line separating two salient objects, but the things would be more complicated and challenging in reality.
2.1.2.3 Lines

Lines and contours play an important role in sending the photographer’s message to the audience. Precisely horizontal, vertical, or diagonal lines show the actual, implied, or even psychic meaning [34]. From a computer visionist’s angle, it is not easy to detect meaningful contour or psychic lines. Still, the detection of the explicit lines is very common as it is easier to analyze. Diagonal lines, the most common usage in aesthetics assessment, divide the image into two overlapping squares with one side on the lower dimension, where bisected region by the diagonals of the square covers the salient part of the image. Figure 2.9 focuses on the position of the body of the guy as the image foreground.

Figure 2.9: Diagonal lines intersect at subject faces [taken from Eric Kim photography website].
2.1.2.4 Triangle

One of the known composition techniques in portrait photography is the triangle method. Human eyes like to see around and imagine the basic geometric shapes from the objects. Inherently, the photos containing such geometric shapes are more appealing than the others. Figure 2.10 shows the triangle technique used by the photographer.

Figure 2.10: A girl creating the triangle technique by her hand gesture.

2.1.2.5 Other techniques

Because the human eyes intuitively search the dominant part of the image, most of the aesthetic photography techniques try to entice the audience’s attention to the salient region. The other methods used by professional photographers include simplicity, vanishing lines, framing, and depth of field (first investigated by Datta et al. [38]).

Figure 2.11: Simplicity technique emphasizes a single foreground object like a dish.

As an example, figures 2.11, 2.12, 2.13, and 2.14 illustrate some of known photography techniques. As you can realize, the first dominant object in figure 2.11 is simply a dish, and the roadsides and other lines in figure 2.12 are vanishing at a
Figure 2.12: Leading lines vanishing at a point illustrates a feeling of a 3D view distant point, and figure 2.13 has been framed by a curved hand to bold a face also using depth of field, and a dandelion among bushes in figure 2.14 is projected and focused sharply by the depth of field technique.

Figure 2.13: Framing technique using two fingers attracts viewer sight to the main subject.

Figure 2.14: Depth of field technique makes the foreground subject sharper than background.
2.2 Related Work

In this section, we summarize the related work in the field of image aesthetics. As linear perspective is one aspect of image aesthetics, we cover vanishing point (VP) detection after that. Furthermore, the proposed work focuses on portrait image aesthetics, and we do a literature survey on human pose detection, portrait aesthetics assessment, and online/offline feedback systems for photography.

2.2.1 Image Aesthetics in Computer Science

The inception of image aesthetics in computer science is from the middle of the last decade. To assess the aesthetics of the images, the photo assessor should output a scale or yes/no answer to compare with other results. The image assessment process is similar to other machine learning fields, but the feature characteristics are different. More related features lead to a more accurate predictor, so the quality of the features matter.

After extracting rules-related features from the image like most evaluation systems, the feature vectors of the training dataset are calculated. Then, the underlying learning framework is trained by these feature vectors, and the needed coefficient would be updated. After the learning phase, a new image can be tested and then rated or classified by the system based on the perceived features and learning algorithm.

2.2.1.1 Image Aesthetics Assessment

Books on professional photography [7–9, 34, 36, 37, 39, 40] guide people to master skills of taking striking photos practically in various situations. However, learning through them takes a lot of time and practice. Existing technical approaches attempt to automatize this process, but they are limited and mostly focused on offline evaluation or active manipulation of the photos after they are taken. Basic image aesthetics and composition rules in visual art [34, 36, 37], including geometry, color palette, and the rule of thirds, have first been studied computationally by Datta et al. [38] and Ke et al. [41]. They start to explore the challenges in inferring image aesthetics by extracting visual features from rated photos. After the initiation of the topic, most of the works try to improve their feature set or
exploit the conventional methods in computer vision to compose a more beautiful photo.

The features used for image aesthetics were mostly low-level and global. There is a common idea in other computer vision fields to use some local features as well. Luo et al. [42] also Wong and Low [43] try to leverage a saliency map method to bold the salient part of the image as the more appealing part of the image. They extract some local features from this salient part and previous global features to train their aesthetic model.

Marchesotti et al. [44] show that generic image descriptors are also very useful to assess image aesthetics, and the performance of the trained model can be improved significantly. They continue their work and build a pretty common dataset for aesthetics assessment called Aesthetic Visual Analysis (AVA) [45] which contains many kinds of rated images from known sharing websites.

Deep learning-based approaches [46–50] exploit customized architectures to train image aesthetic-quality models with annotated datasets, and the outcome is an estimation for actual (average) or personalized [51] aesthetic rating of an image.

2.2.1.2 Image Re-composition

There are some online auto-composition or re-composition systems that actively change the taken photo by the user for a better composition or aesthetics. Bhat-tacharya et al. [52] suggests a framework to improve the visual quality of the taken photo using several spatial re-composition methods. We are interested in categorized these re-composition techniques as follows.

The cropping techniques used in [53–56] separate the region of interest (ROI) with the help of a saliency map or eye fixation. Zhang et al. [57] use aesthetic rules as some templates to automatically crop a digital image. Also, other similar works [58–60] focus on the visual aesthetics features in the salient region to find a better intuition about how to do cropping.

Another type of re-composition, Liu et al. [61] use the warping method to recompose the photo, i.e., representing the image as a triangular or quad mesh, the image maps to another mesh while keeping the semantics and perspectiveness. Also, R2P [62] using triangular mesh and saliency map detects the foreground part
in reference and input image and then, using a graph-based algorithm, re-targets the salient part of the image via the re-composition of the part to the best fitting position.

The other known technique in image re-composition is patch re-arrangement, where a region of interest in image patches to the other region of the image. There are three types of patch re-arrangement. First, pure patch re-arrangement [63–65] detects the group of pixels on the borders of the patch and matches this group to the other vertical or horizontal group of pixels near the patched area. Second, cut and paste methods [52, 66] remove the salient part, re-paint the foreground with respect to salient parts and borders, and then paste it to the desired position in the image. Finally, seam carving [67, 68] cuts the useless seams and replaces them in other positions.

2.2.1.3 Portrait Composition

While there exist prior studies on image aesthetics assessment, few considered portrait photography in-depth, even though the portion of portrait genre is very high in photography. Prior works have not explored a novel method to address the problem in photographic portraiture, beyond combining and using well-known features or modifying features to apply. We categorize prior works into two main groups: rule-based evaluation models [69, 70] that exploit known photography rules to assess portraits, and facial evaluation models [71–75] that use visual features on face like smiling, age, gender, etc.

In a rule-based evaluation model, Khan and Vogel [69] show that a small set of face-centered spatial features extend the rule of thirds and perform better than a large set of aesthetics-related features. Their dataset containing 500 images from Flickr scored by 40 people is limited for a general conclusion. However, their aesthetic features, especially spatial features, are close to well-known photography rules widely investigated before. Males et al. [70] explore the aesthetic quality of head-shots using some famous photography rules and low-level facial features. More specifically, sharpness and depth of field, the rule of thirds, contrast, lightness, hue counts, and face size are exploited as their fundamental features. Unfortunately, the experimental results of the paper are limited, and it is hard to conclude for general cases. Xue et al. [71] study the design inferring portrait aesthetics with
appealing facial features like smiling, orientation, to name but a few. Similarly, Harel et al. [76] exploit traditional features like hue, saturation, brightness, contrast, simplicity, sharpness, and the rule of thirds. They also extract saliency maps by graph-based visual saliency. Then, they calculate the standard deviation and the main subject coincidence of the saliency map. The other facial evaluation models [72, 74, 75] use well-known low-level aesthetic features such as colorfulness, sharpness, and contrast, as well as high-level face-related features such as gender, age, and smile. Their idea is based on exploiting these features for all segmented parts of the face, including hair, face, eyes, and mouth. Redi et al. [73] show that the beauty of the portrait is related to the amount of art used in it, not the subject beauty, age, race, or gender. Using a dataset derived from AVA [45], they exploit a high-dimensional feature vector including aesthetic rules, biometrics and demographic features, image quality features, and fuzzy properties. Based on lasso regression output, eyes sharpness and uniqueness features have the highest rank for a good portrait.

Also, the lighting of the facial photography [77] has been analyzed using templates trained by different lighting styles. Their features are defined as local contrasts in multiple face regions with a distance measure as face illumination descriptor. Mazza et al. [78] want to explore the influential psychological factors on the perceived context of a portrait. Using crowdsourcing, they gather more data to do statistical analysis. However, the results are limited to the relation of “dress to a subject job” and “gender to dating”.

2.2.2 Vanishing Point Detection

Linear perspective giving a 3D feeling to an image counts as an important aesthetic characteristic of an image, and it is captured by vanishing point (VP) detection methods. Most existing VP detection algorithms are based on clustering edges in the image according to their orientations [14, 79, 80]. In [81], Xu et al. studied various consistency measures between the VPs and line segments and developed a new method that minimizes the uncertainty of the estimated VPs. Lezama et al. [82] proposed to find the VPs via point alignments based on the a contrario methodology. Instead of using edges, Vedaldi and Zisserman proposed to detect VPs by aligning self-similar structures [83].
Recently, there is increasing interest in exploiting special scene structures for VP detection. For example, many methods assume the “Manhattan World” model, indicating three orthogonal parallel line clusters [17, 84–86]. When the assumption holds, they are shown to improve the VP detection results. However, such an assumption is invalid for typical natural scenes. Other related studies detect VPs in specific scenes such as unstructured roads [87–89], but it remains unclear how these methods can be extended to general natural scenes.

The classic approaches to content-based image retrieval [90] typically measure the visual similarity based on low-level features (e.g., color, texture, and shape). Recently, thanks to the availability of large-scale image datasets and computing resources, complicated models have been trained to capture the high-level semantics about the scene [91–94]. However, because many visual descriptors are generated by local feature extraction processes, the overall spatial composition of the image (i.e., from which viewpoint the image is taken) is usually neglected. To remedy this issue, [3] first classify images into pre-defined composition categories such as “horizontal”, “vertical”, and “diagonal”. Similar to this work, [95] also explores the VPs in the image for retrieval. However, it assumes known VP locations in all images, thus cannot be applied to a general image database where most images do not contain a VP.

2.2.3 Human Pose Estimation

We believe human pose in portrait photography plays an important role as an aesthetic characteristic. Human pose estimation is the procedure to estimate the human body kinematic graph from a still image. It is a key problem in computer vision for about two decades, as it has many applications like robotics and Kinect-based games. Also, there are many other variants where human pose analysis becomes easier, such as detecting the upper body, tracking a video, and estimating using a single depth image because any information like pixel depth value or frame redundancy can make the problem easier.

However, depth values can help the pose estimation, and many depth cameras have a non-negligible amount of noise, making it hard to estimate as most approaches are based on probabilistic schemes. Body self-occlusion is the main issue in single-view human pose recovery, and it has remained an open problem in a general sense. Even
if we want to estimate hand pose, the problem is not easy as the whole body, as the number of the hand pose configurations are very high. The high degree of freedom (DoF) for body loose-limbed model [96] makes the human pose characterization problem very high dimensional.

Based on the broad family of algorithms, human pose estimation is equivalent to finding the most probable human body pose configuration ($\theta$) from the set of all possible poses ($\Theta$) with knowing some given observation ($O$) such as single depth map, single time instant or activity information, i.e.:

$$\hat{\theta} = \arg \max_{\theta=\Theta} P(\theta|O).$$  \hspace{1cm} (2.1)

where $P(.)$ is the conditional probability function of pose $\theta$ given observation $O$. The argument-max function will go over all possible poses available inside $\Theta$. There are many discriminative algorithms where learning-based algorithms are more famous than non-learning algorithms to solve this problem. Learning-based algorithms can be performed in real-time after lengthy offline processing on a highly varied dataset of training images.

From a single depth image without any temporal information, Girshik et al. [22], and then Shotton et al. [23, 24] from the Microsoft Research Kinect group investigate two frameworks to tackle human pose estimation. In the first approach, they formulate the problem as a labeling problem, and the depth values are related to the body parts. This representation localizes the body joints using a per-pixel classification. In the following approach, body joints positions are captured directly from a random forest regression.

As human pose estimation is improving, the limitations are lifted gradually. In this way, the better training dataset would give better estimation results. Andriluka et al. [97] propose “MPII Human Pose” as a diverse, extensive, and 2D-based benchmark. They have annotated more than 800 different activities with body parts and joints, head orientations, and occlusions. Also, they test the benchmark on some pose estimators showing Pishchulin et al.’s work [98] is better using spatial models and various types of appearance representation.

The estimation accuracy has been improved in most recent works. DeeperCut [25] investigates the articulated body pose estimation with multiple humans. They
improve body part estimators, pairwise proposal assembling, and gradual optimization schemes to converge faster. Also, Wei et al. [26] target a convolutional network architecture for pose machines, and they address the problem of diminishing coefficients during the training stage by changing the cost function appropriately. Furthermore, Newell et al. [27] use another convolutional network architecture for human pose estimation called “Stacked Hourglass”. While having superior accuracy, they repeat the bottom-up processing with proper scales as sequential steps of pooling and upsampling.

2.2.4 On-site Feedback Systems

An aesthetic assessor may find a metric to evaluate the aesthetic quality of an image, but the way it conveys this assessment to take a better photo is also crucial. Because an amateur photographer probably has no idea about how to improve the image composition. That is why providing meaningful feedback to enhance the following shots and not just image aesthetic assessment is one of our main intentions. Giving feedback on a photographic system firstly has been introduced by Joshi et al. [2], as they suggest a real-time filter to trace and aesthetically rate the camera shots, and then the photographer should learn what a better shot is.

The idea of real-time feedback has been taken from Barry’s works [99–101] which is talking about how to convey common knowledge of videography to an amateur via a mindful camera. However, they do not go into the details of the implementation of a photographic system.

In [1] the authors search for views within a scene and retrieve a better crop relying on known structural features and visual saliency of professional photographs. There exist prior works that use the meta-data information (such as geo-location, weather, and time) and recommend a better position in the frame for standing people [102, 103], camera state guidance [104], or quality and uniqueness-aware view cells [105] where their dataset contains known landmark photos.

An on-site aesthetics feedback system [3, 4] retrieves similar images with known composition rules as qualitative feedback. Also, it gives color combination feedback for having colorfulness in the next photo and outputs the overall aesthetic rating of the input photo. OSCAR [3] is assumed to fulfill the future needs of an amateur
photographer. However, giving such feedback may be unrelated or unrealistic to the user, and the retrieved results may not be aesthetically helpful to a photographer.

Xu et al. [106] suggest using a three-camera array to enhance the quality of the taken photos by the rule of thirds. The smartphone interface using the camera array information shows some real-time guidelines for taking a photo from another position. The work starts very well to find an application for future mobile phones with a camera array, but the focus of the work is only on a very well-studied feature, the rule of thirds.

More recently, a perspective-related technique [5] and a triangle technique [6] retrieve similar photos to a query photo having perspective or triangle. But using these common photography rules restricts the coverage and diversity of the photography ideas, while primitives such as human pose, photo genre, or scene semantics are not directly considered. Compared to the previous work in [107], we use a much broader dataset in terms of the size (about twice), diversity (portrait and landscape), quality (higher rating photos) of the dataset (Sections 5.2 and 5.6.1). Further, we implemented novel object detectors in Sections 5.3.1 and 5.6.2, novel category detection (Sections 5.3.2 and 5.6.2.5), novel artistic pose clustering (Sections 5.3.3 and 5.6.2.6). We also employed a novel integration of detectors (hysteresis fusion) in Sections 5.3.1.2 and 5.6.2.4, new decomposition and composition models with a faster retrieval system for user preferences and ranking (Sections 5.4 and 5.6.3). Also, our new method is extensively compared with recent deep learning-based methods for object detection [108–113] and image retrieval [114, 115].
Chapter 3  |  Detecting Vanishing Points in Natural Scenes

3.1 Introduction

Recently, with the widespread use of digital cameras and other mobile imaging devices, there has been increasing interest in the multimedia community in building intelligent programs to analyze the visual aesthetics and composition of photos automatically. Information about photo aesthetics and composition [116] is shown to benefit many real-world applications. For example, it can be used to suggest improvements to the aesthetics and composition of photographers’ work through image re-targeting [117, 118], as well as provide on-site feedback to the photographer at the point of photographic creation [3, 119].

In this work, we focus on an important principle in photo composition, namely, the use of linear perspective. It corresponds to a relatively complex spatial system that concerns primarily with the parallel lines in the scene. Indeed, parallel lines are one of the most prevalent geometric structures in both man-made and natural environments. Under the pinhole camera model, they are projected into image lines that converge to a single point, namely, the vanishing point (VP). Because the VPs provide crucial information about the geometric structure of the scene, automatic
detection of VPs has long been an active research problem in image understanding.

In the literature, existing VP detection methods mainly focus on the *man-made environments*, which typically consist of a large number of edges or line segments aligned to one or more dominant directions. To this end, numerous methods have been proposed to cluster line segments into groups, each representing a VP in the scene [14, 17, 79–82]. These methods have successfully found real-world applications such as self-calibration, 3D reconstruction of urban scenes, and stereo matching.

![Natural scene images with vanishing points. Images are from the “landscape” category of the AVA dataset. Manually labeled ground truth lines are marked in green.](image)

Figure 3.1: Natural scene images with vanishing points. Images are from the “landscape” category of the AVA dataset. Manually labeled ground truth lines are marked in green.

However, little attention has been paid to the *natural landscape scenes*. In natural scene images, a VP is detectable when there are as few as two parallel lines in space. As shown in Figure 3.1, such VPs and the associated geometric structures convey a strong sense of 3D space or depth to the viewers. While human eyes have little difficulty identifying the VPs in these images, automatic detection of VPs poses a significant challenge to computer systems for two main reasons. *First, the visible edges can be weak and not detectable via local photometric cues.* Existing line segment detection methods typically assume the gradient magnitude of an edge pixel is above a certain threshold (e.g., Canny edge detector [120]) or the number of pixels with aligned gradient orientations is above a certain threshold (e.g., LSD [121]). However, determining the threshold can be difficult due to the
weak edges and image noise. *Second, the number of edges converging to the VP may be small compared to irrelevant edges in the same scene.* As a result, even if one can detect the converging edges, clustering them into one group can be demanding due to a large number of outliers.

To overcome these problems, we propose to make use of the image contours to detect edges in natural images. Compared to local edge detectors, image contours encode valuable global information about the scene, thus are more effective in recognizing weak edges while reducing the number of false detections due to textures. By combining the contour-based edge detection with J-Linkage [21], a popular multi-model detection algorithm, our method has been shown to significantly outperform the state-of-the-art methods on detecting the dominant VP in natural scene images.

As an application of our VP detection method, we demonstrate how the detected VPs can be used to improve the usefulness of existing content-based image retrieval systems in providing on-site feedback to amateur photographers. In particular, we note that linear perspective is known as an effective tool in recognizing the viewer as a specific unique individual in a distinct place with a point of view [34]. Therefore, given a photo taken by the user, we study the problem of finding photos about similar scenes and with similar viewpoints in a large collection of photos. These photos can be potentially used to guide the user on his/her work. Further, in this task, we are also the first to answer an important yet largely unexplored question in the literature: *How to determine whether there exists a dominant VP in a photo?* To this end, we design a new measure of strength for a given candidate VP and systematically examine its effectiveness.

In summary, we make the following contributions:

- We are among the first to study the problem of detecting vanishing points in natural landscape scenes. We propose a new method for dominant VP detection. By combining a contour-based edge detector with J-Linkage, our method significantly outperforms the state-of-the-art methods for natural scenes.
- We develop a new strength measure for VPs and demonstrate its effectiveness in identifying images with a dominant VP.
• We demonstrate the application of our method for assisting amateur photographers via viewpoint-specific image retrieval.

• To facilitate future research, we have created and made available a manually labeled dataset for dominant VPs in over 1,300 natural scene images.

3.2 Ground Truth Dataset

To create our ground truth dataset, we leverage the open AVA dataset [45], which contains over 250,000 images along with a variety of annotations. The dataset provides semantic tags describing the semantics of the images for over 60 categories, such as “natural”, “landscape”, “macro”, and “urban”. For this work, we used the 21,982 images labeled as “landscape”.

Next, for each image, we need to determine whether it contains a dominant VP and, if so, label its location. Note that our ground truth data is quite different from those in existing datasets such as York Urban Dataset (YUD) [122] and Eurasian Cities Dataset (ECD) [80]. While these datasets are focused on urban scenes and attempt to identify all VPs in each image, our goal is to identify a single dominant VP associated with the main structures in a wide variety of scenes. The ability to identify the dominant VP in a scene is critical in our targeted applications related to aesthetics and photo composition.

Like existing datasets, we label the dominant VP by manually specifying at least two parallel lines in the image, denoted as $l_1$ and $l_2$ (see Figure 3.1). The dominant VP location is then computed as $v = l_1 \times l_2$. Because our goal is to identify the dominant VPs only, we make a few assumptions during the process. First, each VP must correspond to at least two visible parallel lines in the image. This assumption eliminates other types of perspective in photography, such as diminishing perspective, which is formed by placing identical or similar objects at different distances. Second, for a VP to be the dominant VP in an image, it must correspond to some major structures of the scene and carries more visual weight than other candidates, if any. We do not consider images with two or more VPs carrying similar visual importances, typically seen in urban scenes. Similarly, we also exclude images where it is impossible to determine a single dominant direction due to parallel curves (Figure 3.2). Finally, observing that only those VPs which lie
within or near the image frame convey a strong sense of perspective to the viewers, we resize each image so that the length of its longer side is 500 pixels and only keep the dominant VPs that lie within a $1000 \times 1000$ frame, with the image placed at the center. We used the size 500 pixels as a reasonable compromise between keeping details and providing fast runtime for large-scale applications.

We collected a total of 1,316 images with annotations of ground truth parallel lines. The dataset is publicly available.

![Figure 3.2: Example natural scene images that are not suitable for this work. The first two images show diminishing perspective. The third image has two VPs. The last image contains parallel curves, not parallel lines.](image)

### 3.3 Contour-Based Vanishing Point Detection

Given a set of edges $E = \{E_1, \ldots, E_N\}$, a VP detection method aims to classify the edges into several classes, one for each VP in the scene, plus an “outlier” class. Similar to [79], we employ the J-Linkage algorithm [21] for multiple model estimation and classification. The key new idea of our method lies in the use of contours to generate the input edges. As we will see in this section, our contour-based method can effectively identify weak edges in natural scene images and reduce the number of outliers at the same time, leading to significantly higher detection accuracy.

#### 3.3.1 J-Linkage

Similar to RANSAC, J-Linkage first randomly chooses $M$ minimal sample sets and computes a putative model for each of them. For VP detection, the $j$-th minimal set consists of two randomly chosen edges: $(E_{j1}, E_{j2})$. To this end, we first fit a line $l_i$ to each edge $E_j \in E$ using least squares. Then, we can generate the hypothesis $v_j$ using the corresponding fitted lines: $v_j = l_{j1} \times l_{j2}$. Next, J-Linkage
constructs a \( N \times M \) preference matrix \( P \), where the \((i, j)\)-th entry is defined as:

\[
p_{ij} = \begin{cases} 
1 & \text{if } D(E_i, v_j) \leq \phi \\
0 & \text{otherwise}
\end{cases}.
\] (3.1)

Here, \( D(E_i, v_j) \) is a measure of consistency between edge \( E_i \) and VP hypothesis \( v_j \), and \( \phi \) is a threshold. Note that \( i \)-th row indicates the set of hypotheses edge \( E_i \) has given consensus to and is called the preference set (PS) of \( E_i \). J-Linkage then uses a bottom-up scheme to iteratively group edges that have similar PS. Here, the PS of a cluster is defined as the intersection of the preference sets of its members. Specifically, in each iteration, it computes the Jaccard distance between any two clusters \( A \) and \( B \):

\[
d_J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|},
\] (3.2)

and merges the two clusters with the smallest distance. The operation is repeated until the distance between any two clusters is 1.

**Consistency measure.** We intuitively define the consistency measure \( D(E_i, v_j) \) as the root mean square (RMS) distance from all points on \( E_i \) to a line \( \hat{l} \), such that \( \hat{l} \) passes through \( v_j \) and minimizes the distance:

\[
D_{\text{RMS}}(E_i, v_j) = \min_{l : l \times v_j = 0} \left( \frac{1}{N} \sum_{p \in E_i} \text{dist}(p, l)^2 \right)^{\frac{1}{2}},
\] (3.3)

where \( N \) is the number of points on \( E_i \).

### 3.3.2 Edge Detection via Contours

Because we rely on edges to identify the dominant VP in an image, an ideal edge detection method should have the following properties: (i) it should detect all edges that converge to the true VPs, (ii) the detected edges should be as complete as possible, and (iii) it should keep the number of irrelevant or cluttered edges to a minimum. As we have discussed, local edge-detection methods do not meet these criteria. Instead, a successful method must go beyond local measurements and utilize global visual information.

Our key insight is that to determine if an edge is present at a certain location,
it is necessary to examine the relevant regions associated with it. This insight is motivated by observing that humans label the edges by identifying the physical objects in an image. In addition, based on the level of details they choose, different people may make other decisions on whether to label a particular edge.

Accordingly, for edge detection, we employ the widely-used contour detection method [20], which proposed a unified framework for contour detection and image segmentation using an agglomerative region clustering scheme. We first discuss the main difference between the contours and edges detected by local methods. Then we show how to obtain straight edges from the contours.

**Globalization in Contour Detection.** The contours detected by [20] enjoy two levels of globalization compared to the local methods.

First, as a global formulation, *spectral clustering* has been widely used in image segmentation to suppress noise and boost weak edges. Generally, let $W$ be an affinity matrix whose entries encode the (local) similarity between pixels. This method solves for the generalized eigenvectors of the linear system: $(D-W)v = \lambda Dv$, where the diagonal matrix $D$ is defined as $D_{ii} = \sum_j W_{ij}$. Let ${v_0, v_1, \ldots, v_K}$ be the eigenvectors corresponding the $K+1$ smallest eigenvalues $0 = \lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_K$. Using all the eigenvectors except $v_0$, one can then represent each image pixel with a vector in $\mathbb{R}^K$. As shown in [20], the distances between these new vectors provide a denoised version of the original affinities, making them much easier to cluster.

Second, a graph-based *hierarchical clustering* algorithm is used in [20] to construct an *ultrametric contour map* (UCM) of the image (see Figure 3.3(b)). The UCM defines a duality between closed, non-self-intersecting weighted contours and a hierarchy of regions. Different levels of the hierarchy correspond to different levels of detail in the image. Thus, each weighted contour in UCM represents the dissimilarity of two, possibly large, regions in the image rather than the local contrast of small patches.

**From Contours to Edges.** Let $C = \{C_1, C_2, \ldots\}$ denote the set of all weighted contours. To recover straight edges from the contour map, we apply a scale-invariant contour subdivision procedure. Specifically, for any contour $C_j$, let $c_1^j$ and $c_2^j$ be the two endpoints of $C_j$, we first find the point on $C_j$ which has the maximum
distance to the straight line segment connecting its endpoints:

\[
p^* = \arg \max_{p \in C_j} \text{dist}(p, \overrightarrow{c_j^1c_j^2}). \tag{3.4}
\]

We then subdivide \(C_j\) at \(p^*\) if the maximum distance is greater than a fixed fraction \(\alpha\) of the contour length:

\[
\text{dist}(p^*, \overrightarrow{c_j^1c_j^2}) > \alpha \cdot |C_j| . \tag{3.5}
\]

By recursively applying the above procedure to all the contours, we obtain a set of approximately straight edges \(E = \{E_1, \ldots, E_N\}\). We only keep edges that are longer than a certain threshold \(l_{\text{min}}\) because short edges are susceptible to image noises (Figure 3.3(c)).

### 3.3.3 Experiments

In this section, we present a comprehensive performance study of our contour-based VP detection method, and compare it to the state-of-the-art. Similar to previous work (e.g., [17, 79]), we evaluate the performance of a VP detection method based on the consistency of the ground truth edges with the estimated VPs. Specifically, let \(\{E_k^G\}_{k=1}^K\) be the set of ground truth edges, the consistency error of a detection \(\hat{v}\) is:

\[
\text{err}(\hat{v}) = \frac{1}{K} \sum_k D_{RMS}(E_k^G, \hat{v}) . \tag{3.6}
\]

For all experiments, we compute the average consistency error over five independent trials.
3.3.3.1 Comparison of Edge Detection Methods

We first compare our contour-based edge detection to the popular Canny detector [120] and LSD [121] in terms of the accuracy of the detected VPs. For our contour-based method, the parameters are: $\alpha = 0.05$, $l_{\text{min}} = 40$, and $\phi = 3$. For the Canny detector and LSD, we tune the parameters $l_{\text{min}}$ and $\phi$ so that the highest accuracy is obtained. In this experiment, we keep the VP with the largest support set as the detection result and report the accuracy in Figure 3.4(a). As one can see, our contour-based method significantly outperforms the other edge detection methods.
Figure 3.5: Comparison of different edge detection methods. The four rows show the original images, and the edges detected by Canny detector, LSD, and our contour-based method, respectively. Yellow edges indicate the edges consistent with the ground truth dominant vanishing point.

In Figure 3.5, we further show some example edge detection results. Since most VP detection methods rely on clustering the detected edges, an ideal edge detector should maximize the number of edges consistent with the ground truth dominant VP and minimize the number of irrelevant edges. As shown, our contour-based method can better detect weak yet important edges in terms of both quantity and completeness. For example, our method can detect the complete edges of the road in Figure 3.5(b), while the local methods only detected parts of them. Also, only our method successfully detected the edges of the road in Figure 3.5(c).

Another important distinction between our contour-based method and the local methods concerns the textured areas in the image. Local methods tend to confuse image texture with true edges, resulting in a large number of detections in these areas (e.g., the sky region and the road in 3.5(d) and (e), respectively). Such false positives often lead to incorrect clustering results in the subsequent VP detection stage. Meanwhile, our method treats the textured area as a whole, significantly reducing the number of false positives.
3.3.3.2 Comparison with State-of-the-Art

Next, we compare our contour-based method to the state-of-the-art VP detection methods. As we discussed before, most existing methods focus on urban scenes and make strong assumptions about the scene structures, such as a Manhattan world model [17, 84–86]. Such strong assumptions render these methods inapplicable to natural landscape scenes.

While other methods do not explicitly assume a specific model, they still benefit from the scene structures to various extents. In Figure 3.4(a), we compare our method to two recent methods, namely Tretiak et al. [80] and Lezama et al. [82]. Note that [82] uses the Number of False Alarms (NFA) to measure the importance of the detected VPs. For a fair comparison, we keep the VP with the highest NFA. Figure 3.4(a) shows that the two methods do not perform well on the natural landscape images. The problem with [80] is that it assumes multiple horizontal VP detections for the horizon and zenith estimation, but there may not be more than one VP in natural scenes. Similarly, [82] relies on the multiple horizontal VP detections to filter redundant and spurious VPs.

3.3.3.3 Parameter Sensitivity

We further study the performance of our contour-based VP detection method w.r.t. the parameters $\alpha$, the minimum edge length $l_{\text{min}}$, and the distance threshold $\phi$ in Eq. (3.1). We conduct experiments with each parameter while the others are fixed. The default parameter setting is $\alpha = 0.05$, $l_{\text{min}} = 40$, and $\phi = 3$.

*Performance w.r.t. $\alpha$.* Recall from Section 3.3.2 that $\alpha$ controls the degree to which a contour segment may deviate from a straight line before it is divided into two sub-segments. Figure 3.4(b) shows the best performance is achieved with $\alpha = 0.05$.

*Performance w.r.t. minimum edge length $l_{\text{min}}$.* Figure 3.4(c) shows the performance of our method as a function of $l_{\text{min}}$. Rather surprisingly, we find that the accuracy is quite sensitive to $l_{\text{min}}$. Because the number of edges consistent with the dominant VP is relatively small for natural scenes. Therefore, if $l_{\text{min}}$ is too small, these edges may be dominated by irrelevant edges in the scene; if $l_{\text{min}}$ is too large, there may not be enough inliers to estimate the VP location robustly.
Performance w.r.t. threshold $\phi$. Figure 3.4(d) shows the accuracy of our method w.r.t. the threshold $\phi$ in Eq. (3.1). As one can see, our method is relatively insensitive to the threshold, and achieves the best performance when $\phi = 3$.

### 3.4 Selection of the Dominant Vanishing Point

In real-world applications concerning natural scene photos, it is often necessary to select the images in which a dominant VP is present since many images do not have a VP. Further, if multiple VPs are detected, we need to determine which one carries the most important photo composition. Therefore, given a set of candidates $\{v_j\}_{j=1}^n$ generated by a VP detection method, our goal is to find a function $f$ which well estimates the strength of a VP candidate. Then, we can define the dominant VP of an image like the one whose strength is (i) the highest among all candidates and (ii) higher than a certain threshold $T$:

$$v^* = \arg \max_{f(v_j) \geq T} f(v_j). \quad (3.7)$$

In practice, given a detected VP $v_j$ and the edges $E_j \subseteq \mathcal{E}$ associated with the cluster obtained by a clustering method (e.g., J-Linkage), a simple implementation of $f$ would be the number of edges: $f(v_j) = |E_j|$. Note that it treats all edges in $E_j$ equally. However, we have found that this is problematic for natural images because it does not consider the implied depth of each edge in the 3D space.

#### 3.4.1 The Strength Measure

Intuitively, an edge conveys a strong sense of depth to the viewers if (i) it is long, and (ii) it is close to the VP (Figure 3.1). This observation motivates us to examine the implied depth of each point on edge instead of treating the edge as a whole.

Geometrically, as shown in Figure 3.6, let $E$ be a line segment consistent with vanishing point $v = (v_x, v_y, 1)^T$ in the image.$^2$ We further let $D$ be the direction in 3D space (i.e., a point at infinity) that corresponds to $v$: $v = PD$, where $P \in \mathbb{R}^{3 \times 4}$ is the camera projection matrix.

$^2$In this section, all 2D and 3D points are represented in homogeneous coordinates.
Figure 3.6: Illustration of our edge strength measure.

For any pixel on the line segment \( q = (q_x, q_y, 1)^T \in E \), we denote \( Q \) as the corresponding point in the 3D space. Then, we can represent \( Q \) as a point on a 3D line with direction \( D \): \( Q = A + \lambda D \), where \( A \) is some reference point chosen on this line, and \( \lambda \) can be regarded as the (relative) distance between \( A \) and \( Q \). Consequently, we have

\[
q = PQ = P(A + \lambda D) = a + \lambda v ,
\]

where \( a = (a_x, a_y, 1)^T \) is the image of \( A \). Thus, let \( l_q \) and \( l_a \) denote the distance on the image from \( q \) and \( a \) to \( v \), respectively, we have

\[
\lambda = l_a/l_q - 1 .
\]

Note that if we choose \( A \) as the intersecting point of the 3D line corresponding to \( E \) and the image plane, \( \lambda \) represents the (relative) distance from any point \( Q \) on this line to the image plane along direction \( D \). In practice, although \( l_a \) is typically unknown and varies for each edge \( E \), we can still infer from Eq. (3.9) that \( \lambda \) is a linear function of \( 1/l_q \), that motivates us to define the weight of a pixel \( q \in E \) as \( (l_q + \tau)^{-1} \), where \( \tau \) is a constant chosen to make it robust to noises and outliers. Thus, our new measure of strength for \( v_j \) is defined as

\[
f(v_j) = \sum_{E \in \mathcal{E}_j} \sum_{q \in E} \frac{1}{l_q + \tau} .
\]

Edges that are longer and closer to the VP have more weights, according to our new measure.
3.4.2 Experiments

3.4.2.1 Dominant Vanishing Point Selection

We first demonstrate the effectiveness of the proposed strength measure in selecting the dominant VP from the candidates obtained by our VP detection algorithm. In Figure 3.7(a), we compare the following three measures in terms of the consistency error of the selected dominant VP:

Edge Num: The number of edges associated with each VP.

Edge Sum: The sum of the edge lengths associated with each VP.

Proposed: Our strength measure Eq. (3.10).

As shown, by considering the length of an edge and its proximity to the VP, our proposed measure achieves the best performance in selecting the dominant VP in the image.

3.4.2.2 Dominant Vanishing Point Verification

Next, we evaluate the effectiveness of the proposed measure in determining the existence of a dominant VP in the image. For this experiment, we use all the 1,316 images with labeled dominant VPs as positive samples and randomly select 1,500
images without a VP from the “landscape” category of the AVA dataset as negative samples. In Figure 3.7(b), we plot the ROC curves of the three different measures. As a baseline, we also include the result of the Number of False Alarms (NFA) score proposed in [82], which measures the likelihood that a specific configuration (i.e., a VP) arises from a random image. One can see that our proposed measure achieves the best performance.

In Figure 3.8(a) and (b), we further plot the percentages of images as a function of our strength measure and the average consistency error, respectively. In particular, Figure 3.8(b) shows that the consistency error decreases substantially when the strength score is higher than 150. Thus, our strength measure is a good indicator of the reliability of a VP detection result.

Figure 3.8: The impact of VP strength on the accuracy of dominant VP detection.
Chapter 4 | Toward Assisting with Portrait Composition

4.1 Introduction

Art still has many ambiguous aspects out of the known sciences, and the beauty of the art comes from the virgin novelty by artists. It is still daunting for a machine to compose an impressive original song, painting, or script. However, high-resolution photography has been made ubiquitous by recent technologies, such as high-quality smart camera phones. Also, the aesthetics of photography is known as a collection of rules in artistic literature [34, 36, 37] such as balance, geometry, symmetry, the rule of thirds, and framing. Digital photography is of great interest among most people using social networking and photo-sharing websites such as Facebook, Google Photos, Twitter, and Instagram. However, getting a striking photo involves experience and skills and is often not easy.

One of the key goals of photography is to take impressive and memorable photos. To achieve this goal, the photographer should study many advanced techniques and experimentally practice many different situations. However, it is very costly for beginners to learn and perform these methods, and it takes a while to become professional in understanding the image aesthetics.

As mentioned before, there are many approaches to assess the image quality reviewed in Section 2.1.2 based on the known rules in advanced photography such
as the rule of thirds, the triangle, balancing elements, golden ratio, leading lines, and diagonals. Also, based on these rules, image re-targeting, auto-cropping, and scaling, and even auto foreground object replacement have been proposed to capture better photos from the scene mostly.

Complying with these photography rules gives a better photo composition, as we have exploited perspective-related rules [5], but these techniques are not sufficient. They describe image aesthetics in a very general sense that sometimes it is impossible to meet all of them together, e.g., having a symmetrical balance with the rule of thirds in a portrait is impossible, as they contradict each other. Also, any manipulation of the taken image, like image re-targeting and re-composing techniques, changes the integrity of the original image, and the photographer might not desire it.

In the remainder of this chapter, we explain multiple contributions that provide an intelligent portrait composition framework. First, we describe the method to accelerate image dataset processing on a cluster. Second, we describe our portrait composition model, which helps people get meaningful feedback for their portrait photography experience. Third, we exploit the triangle technique on our portrait dataset and compare it with some other methods to show the efficiency of the method.

4.2 Accelerating Dataset Processing in Distributed Systems

Nowadays, the image dataset is getting larger and larger having millions of images. Not only annotating such a dataset is time-consuming, but also the training process with various configurations is taxing. The scalability of image dataset processing is a real issue because proper computer architecture (i.e. CPU or GPU) to handle the training jobs is pricey for an individual. If you want to share the resource among a group, then automatic scheduling is an issue. Also, the current operating systems have no decentralized distributed scheduling scheme, and they cannot optimize the whole system for all submitted jobs. Also, job optimization should not be a free option because it is intensive and subtle, while the jobs are coming asynchronously. To investigate the scalability challenges in intensive image processing, we need to
know the available entities as follows:

- Computer architecture (i.e. CPU or GPU)
- Operating system (OS) scheduler
- Dataset (i.e. the number of jobs and the number of images per dataset)
- Algorithm, application or job (the same names throughout this section)

### 4.2.1 Scheduling Challenges

Principal investigators (PI) submit some proposals to get computational slots from large shared systems. However, the available systems have a centralized scheduler, and it is very challenging to schedule the jobs optimally. Some of the sought challenges to leverage such systems are listed as follows:

- Scalability of the framework
- Time-varying jobs
- Fairness across the jobs
- Asynchronous job submissions
- Being an economic system
- Diversity across software/hardware
- Dynamism of the whole system
- System maintenance

Every resource (e.g. project money, deadline, and equipment) has a limit. Scalability is an issue when the limit is close to reaching. We should evaluate the amount of needed computation in terms of instructions per cycle (IPC) and manage to accomplish it in a fair amount of time. Otherwise, we may miss a deadline, and the whole plan will be useless. Job profiling is necessary to estimate the job runtime and derive the behavior during its runtime. Profiling is not a light task because it needs the whole history of one complete run or online monitoring of the job. In addition, the performance history of jobs may change over time because of new architecture, new configuration, and new workload. In addition, it is not efficient
to make the OS monitor all activities of the jobs, not only because it is a very
time-consuming process but also because there may be some security issues in some
scenarios.

When the number of jobs is getting higher, not only the scheduling of them is
difficult, but also keeping fairness across them is very challenging. By definition,
they should be generally treated without any discrimination. However, it is hard
to define a holistic metric across all jobs and try to make it equal for all of them.
Because the applications may have different needs \( e.g., \) one needs more CPU time,
and the other wants more GPU time. Also, these jobs do not come into the system
at a fixed rate. The distribution of the inter-arrival times affects the utilization
of the system, and the system utilization affects the job performance from the
submission time to the completion time.

We may buy a pricey system for every job to optimize its performance, but it
is very costly because there is a budget limit. Thus, the provided system should
be cost-efficient in terms of price and usefulness. Also, not only many software
applications but also many hardware parts should be considered for this purpose,
and it requires some expert to decide about the current edge of the technology.
These jobs come and complete 24 hours a day and seven days a week. Not only the
inter-arrival times but also completion times and inter-departure times matter. We
should monitor system utilization all the time and automatically send feedback to
the system scheduler. There should be some mechanism to revive the system at
the time of blackout, blockage, or any maintenance issue.

### 4.2.2 Our Scenario

The centralized approach, in our opinion, is not efficient for all applications, but
we had no other commercial choice to leverage. We have suggested an auction-based
framework where any new application can participate in a new auction and bid
for its desired resource vector \([123]\). Most of the time, our scenario to execute our
jobs includes a large image dataset (from thousands of images), intensive image
processing algorithm (\( i.e. \) run about 5min on a regular CPU or GPU) as the
bottleneck of the application flow, and a multi-server system. The question is how
we can optimize the runtime of the job to complete as soon as possible.
4.2.3 Randomized Inter-Server Scheduling

When the resources are limited in our scenario, the challenges will shape. We start with computer memory, as the needed memory by our image processing application cannot be fitted into a single node cache. It should reside in the memory of one single node. Thus, the parallelism level cannot be intra-node but inter-node.

We have selected the parallelism across the servers of the system using the network file system (NFS). Each server can access the code and the dataset as inputs via NFS. Because the inputs are not changing during the execution, we do not have any consistency issue at the source. However, the resulted output should be separated for each image. The most trivial approach is to divide the images into “some” equal-size groups for load balancing and schedule each group of images on a different server. This “some” can be 2, 5, 10, or even 100 and 1000. It specifies the order of the parallelism in our scheduling. We can make it low for easier management, but it takes a while to finish. Higher-order of parallelism creates another challenge to synchronize the completion time of the jobs assigned to the groups. This problem is known as a fork-join-based scheduling problem [124] which is also a challenging problem.

Our final solution, called randomized inter-server scheduling (RISS) with floating parallelism, exploits the “floating” number (say $N$ which is an integer, not a floating-number) of servers to execute the algorithm on $N$ different images simultaneously. When a job is scheduled on a server by OS, our scheduler module performs and searches randomly to find an unprocessed image by checking its availability in the completed processes and running processes. If it is completed or another process is working on it, our intra-job scheduler tries another randomly picked image. If the image process is neither completed nor running, the scheduler creates a temporary file showing the image process state as running and then assigns the image to be processed. To solve the consistency at the output, we leverage this mechanism to lock on the running image process.

The main advantage of our scheduler is that it can extend the order of the parallelism as much as the OS scheduler lets to the user limit. When a job is killed or done because of any reason, no maintenance is required to handle the rest of the images because the other live jobs will grab the images eventually.
Algorithm 1: Randomized Inter-Server Scheduling

**Data:** Pseudo-random number generator, Dataset Path, and Result Path.

**Result:** The processed images.

1. Enumerate the files in Dataset Path as a list $L$.
2. Get the number of the files $m$ in the list $L$.
3. Generate a list $R$ containing the permutation of the numbers between 1 and $m$.
4. For all numbers between 1 and $m$ do:
   5. Get the path of $R[i]$-th image in the list $L$.
   6. Check the existence of the image output in Result Path as an output file or temporary file. If it exists, go to step 4; otherwise, continue.
   7. Save temporary file showing the image process is running.
   8. Process the image in other codes.
   9. Save the result of the processing of the image in other codes.
10. Delete the temporary file.
11. Go to step 4 for the next randomly picked image.

The order of the parallelism is floating depends on the number of servers are running the algorithm. The scheduler independently works on any system, and it is architecture/OS/dataset/algorithm-agnostic, *i.e.*, it does not need to do any setup in the mentioned entities. Algorithm 1 describes the pseudo-code of our intra-job scheduler. We have used RISS in all of the experiments used in this dissertation. Also, we leverage RISS to process the images in the dataset of radar images to generate preliminary results for these two works in skeleton matching [125, 126].

### 4.3 Intelligent Portrait Composition Assistance

While there are many styles for photography [37, 40] around the world, selecting proper *photography ideas* for a given scene remains a challenging problem and yet to be adequately investigated. The major problem with taking a good portrait photo in a given location is the lack of a local professional photographer to capture a good portrait pose. Professional photographers usually have expertise and creativity in making good positions intuitively [7–9]. Through reading books about photography, one can get familiar with some common composition rules such as balancing, framing, and the rule of thirds. However, it can still be difficult to select and apply techniques for making genuine photos, in a way similar to the gap between reading
In this work, we focus on a framework that helps people make a better shot for their portrait photos with regard to their current location. Given a prior shot from the photographer or the camera viewfinder, our portrait composition assistance outputs some highly-rated prior-composed photos as assessed feedback. Figure 4.1 shows some highly-rated portrait poses, many taken by professionals, collected from the 500px website, and selected by our framework. These 20 portraits are captured in various locations and scenes. Each of them has its photography idea(s), such as a woman with hat (1st image) has made an apropos pose at the heart of the leading lines (fence), or a woman is sitting with crossed ankles bent legs (4th image) where this pose creates a nice S-shape. These techniques are believed to make portrait photography more appealing. Specifically, we address aesthetic retrieval and evaluation of the human poses in portrait photography and improve the quality of the next shot by providing meaningful and constructive feedback to an amateur photographer.

Figure 4.2 shows the flowchart of our approach to assist an amateur photographer in getting a better shot. Based on the first shot as a query, some highly-rated well-posed results are retrieved from our designated dataset using a portrait aes-
thetic model containing the dataset features. The results are illustrated to the photographer to help compose a better shot. The last shot is captured when the current pose is matched with one of the results closely. The details of the flowchart have been explained in later sections. The main contributions are as follows:

- Extraction of new aesthetic features related to special content such as portrait photography.
- A framework to learn from extended aesthetic features.
- Construction of an aesthetic parametric model of the taken image.
- Optimization of the parametric model distance from the aesthetic ground truth subject to semantic dynamicity constraints.
- An iterative algorithm to find the best solution.
- Qualitative evaluation of a parametric aesthetic model.

In the following sections, we cover our proposed requirements, methodology, implementation, and performance assessment to help an amateur photographer taking a more appealing portrait photo.

### 4.3.1 Requirements

In this section, the requirements of the work are considered from multiple angles. It is assumed that we concentrate on special content, but we aim to continue this work on other content.

- **Content**: Portrait, landscape, close-up, and selfie.
- **Dataset**: AVA (dpchallenge.com), photo.net, and flickr.com.
- **Ground-truth**: labeling pose, vanishing lines and points, also aesthetic rating.
- **Features**: generic image features, rule-based photography features, aesthetics-preserving features with hierarchical categorization.
Figure 4.2: Schematic diagram of our proposed framework: The method consists of image aesthetics extensive learning (iAEL) to learn aesthetic reference features, image parametric modeling (iPAM) to capture image aesthetics dynamicity, and image aesthetic improvement (iAI) to optimizes and recommends the best feedback to photographer.

- **Algorithms:** human pose detection, VP detection, face detection, and triangle detection.
4.3.2 Methodology

In this section, we explain different steps of our algorithm to achieve the goal of helping an inexpert photographer. The main steps of our algorithm are as follows and will be described in the implementation section:

- **Step 0.** Extracting any low-to-high level features from taken images such as face and body landmarks, human pose in portraits, vanishing points and leading lines in landscapes, Segmentation map, and Saliency map.

- **Step 1.** Learning portrait aesthetics model from professional portrait training dataset called image aesthetics extensive learning (iAEL).

- **Step 2.** Constructing image parametric aesthetic model (iPAM):
  - **Step 2-a.** Parametrizing feature vector in terms of aesthetics facts such as human poses, triangles, balance, and rule of thirds.
  - **Step 2-b.** Finding the boundaries of doable actions in the taken image, e.g., shift, rotation, and body movement.

- **Step 3.** Optimizing the distance of the aesthetic parametric model in (2-a) from the learned aesthetics facts model in (0) subject to feasible actions in (2-b) that we call image aesthetics improvement (iAI).

- **Step 4.** Selecting the best solution(s) from (3). If there is no solution, we relax the least important aesthetic fact and iteratively go to (2).

- **Step 5.** Assessing the performance of our system.

4.3.3 Implementation

This section describes the main steps in our methodology section in more detail.

4.3.3.1 Low-to-High Level Features Extraction

To learn an aesthetic fact model or to classify an image, we need to extract a bunch of low-level features and high-level visual features from the image. We believe that the following low-level features of the image and the detected human parts are necessary to extract.
**Pose-related Features:** After pose-annotation in ground truth process or pose-estimation by some human pose estimation algorithms, we can extract pose-related features, which are the relative positions of the human body joints (pelvises, scapulas, clavicula, head, elbows, wrists) or parts (trunk, head, neck, hands, arms, forearms, thighs and shins).

Preferably, we would like to start from the position of the neck bottom \( J_{1,1} \), scapulas \( J_{1,2} \) and \( J_{1,3} \), and the pelvises as \( J_{1,4} \) and \( J_{1,5} \) as they are on the plane of the trunk. As the head \( J_{2,1} \) is connected to the neck, the head position can be relatively expressed by the neck. Also, The elbows \( J_{2,2} \) and \( J_{2,3} \) can be recognized by a length and an angle from scapulas \( J_{1,2} \) and \( J_{1,3} \). The knees \( J_{2,4} \) and \( J_{2,5} \) can be similarly recognized by a length and an angle from pelvises \( J_{1,4} \) and \( J_{1,5} \). Respectively, the wrists \( J_{3,2} \) and \( J_{3,3} \) and the ankles \( J_{3,4} \) and \( J_{3,5} \) are similarly calculated by a length and an angle from elbows and knees. So, we can always calculate the absolute position using 2D polar coordinates as follows:

\[
\forall i, j : J_{i,j} = J_{i-1,j} + r_{i-1,i,j} e^{i\theta_{i-1,i,j}}, \tag{4.1}
\]

where \( r_{i-1,i,j} \) is the length from joint \( J_{i-1,j} \) to joint \( J_{i,j} \) known as a body part, \( \theta_{i-1,i,j} \) is the angle of the line from joint \( J_{i-1,j} \) to joint \( J_{i,j} \) with the horizon, and \( i \) is the unit imaginary number. Note that for a human body \( r_{i-1,i,j} \); \( \forall i, j \) are fixed, but \( \theta_{i-1,i,j} \); \( \forall i, j \) can be changed to some fixed extents. Also having 3D pose-annotated/estimated single depth images, we can calculate the relative 3D position of the joints. In this case, we have:

\[
\forall i, j : J_{i,j} = J_{i-1,j} + r_{i-1,i,j} \Delta(\theta_{i-1,i,j}, \phi_{i-1,i,j}), \tag{4.2}
\]

where \( \theta_{i-1,i,j} \) is polar angle as before and \( \phi_{i-1,i,j} \) is azimuthal angle. Also, \( \Delta(\theta_{i-1,i,j}, \phi_{i-1,i,j}) \) function using spherical coordinates is derived from the following equations:

\[
\Delta(\theta, \phi) = i \sin \theta \cos \phi + j \sin \theta \sin \phi + k \cos \theta, \tag{4.3}
\]

where \( i, j, \) and \( k \) are respectively unit directions in Cartesian coordinates as \( (1, 0, 0) \), \( (0, 1, 0) \), and \( (0, 0, 1) \). Note that the degree of azimuthal angle freedom (the domain of \( \phi_{i-1,i,j} \)) is zero for some joints like elbows and knees. So, we have such action
boundaries for joints as follows:

\[ \forall i, j : \theta_{i-1,i,j}^{\min} \leq \theta_{i-1,i,j} \leq \theta_{i-1,i,j}^{\max}, \quad (4.4) \]

\[ \forall i, j : \phi_{i-1,i,j}^{\min} \leq \phi_{i-1,i,j} \leq \phi_{i-1,i,j}^{\max}, \quad (4.5) \]

As a result, a human body pose (hbp) is represented by:

\[ \text{hbp}(J) = \{ \forall i, j : J_{i,j} \}, \quad (4.6) \]

\[ \text{hbp}(J_1, \theta, \phi) = \{ \forall i, j : J_{1,j}, \theta_{i-1,i,j}, \phi_{i-1,i,j} \}, \quad (4.7) \]

where hbp(J) is our pose-related feature, and the distance from the aesthetic reference pose is derived from 4.10.

**Rule-based and High-level Features:** Also, we need to extract the high-level and rule-based features. These features are mostly about spatial composition, triangle composition, and statistical features as follows:

- Spatial composition features are the location and the intensity of the points of the saliency map of the image. They can be compared with the reference feature from the rule of thirds and golden ratio rule from our distance metric 4.10, where intensity can be seen as a discrete distribution, and the reference feature is the points of the power lines with uniform distribution.

- Triangle composition features including leading lines and vanishing points are introduced by Zhou et al. [5].

- Standard deviation of the saliency map, segmentation map, and other color and texture properties of the image can be calculated as well.

**Generic Aesthetics Features:** There are some known general-purpose aesthetic features of the image in color, texture, and statistics domain as follows:

- Color: Hue, Saturation, Value (HSV), Luminance (Y) can be similar to the features exploited in works [69, 127, 128].

- Texture: Sharpness, Contrast, Homogeneity, Hough Peaks are also used in [69].

**Feature Distance Metric:** To order the features from different samples aestheti-
cally, we need a metric to calculate the distance of the features from the reference. Without loss of generality, we prefer to exploit a modified version of Wasserstein distance or Mallows distance (EMD) as used in PD2-clustering [129, 130], because it is a generalization of a distance metric between any two unequal-length weighted tuples and it can measure the distance between any two bags of words (BoWs) under a discrete distribution. For example, if we assume the center of the mass of the saliency map from power points as the rule of thirds feature, the distance metric is reduced to a simple Euclidean distance since there is no distribution (let say uniform distribution) over all elements. Then, for the distance of two $R_i$-based feature set, we have:

$$D_{R_i}(f_{i,j}, f_{i,k}) = \left( \inf E\{d(f_{i,j}, f_{i,k})^p\} \right)^{\frac{1}{p}}, \quad (4.8)$$

where $R_i$ is the $i$-th rule, and $f_{i,j}$ is the $i$-th rule-based feature set of the $j$-th sample, $p$ (usually 1 or 2) is the order of Wasserstein metric, $\inf(.)$ function is defined over all joint discrete couplings of these two features, $E\{\cdot\}$ is the expected value function, and $d(\cdot, \cdot)$ (usually L1-norm or L2-norm) is the $L_q$-norm function of two equal-length tuples as follows:

$$d(f_{i,j}, f_{i,k}) = \|f_{i,j} - f_{i,k}\|_q, \quad (4.9)$$

where $q$ is the order of the norm called $L_q$-norm. Now, if we expand the expectation for two feature sets, we have:

$$D_{R_i}(f_{i,j}, f_{i,k}) = \left( \inf \sum_{m=1}^{S_m} \sum_{n=1}^{S_n} \|f_{i,j}^{(m)} - f_{i,k}^{(n)}\|_q \gamma_{m,n} \right)^{\frac{1}{p}}, \quad (4.10)$$

where $S_m$ and $S_n$ are respectively the size of $f_{i,j}$ and $f_{i,k}$, $\gamma_{m,n}$ are the set of all couplings of these two features $\forall m \in \{1, \ldots, S_m\} \forall n \in \{1, \ldots, S_n\}$, and $f_{i,j}^{(m)}$ is $m$-th feature in $f_{i,j}$ feature set. More accurately eq. 4.10 is subject to the following constraints:

$$\gamma_{m,n} > 0, \forall m \in \{1, \ldots, S_m\}, \forall n \in \{1, \ldots, S_n\}, \quad (4.11)$$

$$\text{Prob}(f_{i,j}^{(n)}) \geq \sum_{m=1}^{S_m} \gamma_{m,n}, \forall n \in \{1, \ldots, S_n\}, \quad (4.12)$$
\[
\text{Prob}(f_{i,j}^{(m)}) \geq \sum_{n=1}^{S_n} \gamma_{m,n}, \forall m \in \{1, \ldots, S_m\},
\]

\[
\inf(\sum_{m=1}^{S_m} \text{Prob}(f_{i,j}^{(m)}), \sum_{n=1}^{S_n} \text{Prob}(f_{i,k}^{(n)})) = \sum_{m=1}^{S_m} \sum_{n=1}^{S_n} \gamma_{m,n}.
\]

As a concrete example for spatial composition feature mentioned in 4.3.3.1, \(f_{i,j}\) is the points of the saliency map the image, where the number of the points is \(S_m\), and \(\text{Prob}(f_{i,j}^{(m)})\) can be derived from the intensity of the point at \(m\)-th position. The reference feature (say \(f_{i,k}\)) for spatial composition is the points on the power lines dividing the image into 9 equal pieces. Similarly the number of the points on power lines is \(S_n\), and \(\text{Prob}(f_{i,k}^{(n)})\) is uniform \(\forall n\). Therefore, the distance of the extracted feature \((f_{i,j})\) from the rule-based reference feature (say \(f_{i,k}\)) can be derived by 4.10. Similarly we can have a concrete example for pose-related feature distance from reference poses.

### 4.3.3.2 Image Aesthetics Extensive Learning (iAEL)

What differentiates professional photography from amateur one is not a yes/no question. Many factors are mattering to evaluate an artistic work from an inexpert one. While the borders of the factors are not completely clear, finding the total assessment metric even makes the evaluation process more ambiguous. Content-based focus can clarify the darkness of the assessment better, as it is not supposed to compare a landscape with a portrait, but we assess a portrait and suggest a better pose or location to the photographer.

Although basic visual primitives such as color, texture, shape, depth, and saliency play an important role in projecting notions of photography professionalism, these rules are limited to capture all aesthetic aspects of a content-specific photographic composition, such as portrait aesthetics. Our approach extensively focuses on aesthetics features of content-specific photography, as we assume the content category can be automatically recognized by the camera or manually adjusted by the user. Our hierarchical content-based approach can ease the problem, and without the loss of the generality of our approach, we focus on portrait photography.

This section aims to learn on an input training dataset to adjust the weights of the rule-based aesthetic features to assess the test dataset accurately. We propose
two strategies to do aesthetics learning. In both strategies, an aesthetic image dataset including professional photos taken by advanced photographers is needed. In portrait case, the people images with different poses are collected and annotated, including face, trunk, arms, forearms, hands, legs (thighs and shins) with an overall rating to generate our ground-truth dataset. Figure 4.3 shows a sample image annotated as a ground truth.

**Learning directly from the Rules:** To learn directly from $M$ known rules, we quantify and formulate each rule ($R_i \forall i \in \{1, ..., M\}$) as some reference features and try to construct the aesthetic fact model based on these rules or reference features. The reference features are directly inferred from the rule when the rule is straightforwardly defined by some math expressions, or it can be indirectly captured by some annotated professional images as the reference images of the rule.

Deviation of a $R_i$-based feature of $j$–th sample image ($f_{i,j} \forall i \in \{1, ..., M\} \forall j \in \{1, ..., N\}$) from the rule reference is measured by a distance metric (like 4.10). For example, in the case of “rule of thirds”, we have to define a distance metric from the lines and points dividing the photo into nine equal pieces. In the case of having “balance”, it can be defined as the infimum of the distance from a line or point over all possible symmetry lines and points.

After defining the distance metric between the reference feature set and the
perceived feature set from an image, we have to find the weights of the rules in an aesthetic assessment. Therefore, we perform supervised learning [131] on all aesthetic rules to find the importance of each one using ground truth dataset and their ratings.

We have to minimize the risk function of the aesthetic assessment, which is the linear summation of the total loss (discrepancy) between “ratings” and “weighted distance of perceived features from rules”. The solution of this supervised learning is the set of weights (discrete importance map) \( \{w_1, w_2, ..., w_M\} \). Mathematically we have:

\[
\text{Risk}_{\text{aesthetics}}(w_1, w_2, ..., w_M) = \sum_{j=1}^{N} \text{Loss}(r_j, \sum_{i}^{M} w_i D_{R_i}(R_i, f_{i,j})), \quad (4.15)
\]

where \( w_i \) is the weight of importance factor of the i-th rule, and \( r_j \) is the rating of the j-th sample, \( \text{Loss(,)} \) is the loss function like L1-norm or L2-norm, and \( \text{Risk(,)} \) is our cost or risk function to optimize.

**Learning from the Expansion of the Rules:** Learning directly from the rules is easier in aesthetics assessment but not accurate, so we introduce the latter new way of learning in the domain of portrait photography. In this strategy, we try to extract expanded rule-based features from professional photos. Having such features is to explore the discrete importance map of the rule instead of only the peak of the rule.

For example, the rule of thirds is not always a fact in portrait photography and depends on the pose of the subject. In spatial composition case, we can imagine four horizontal and four vertical lines dividing the image into 25 equal pieces (more than the rule of thirds) and try to learn the model based on the importance of the relation between these pieces. We investigate and extend rule-based features to a superset called expanded rule-based features, which cover all aesthetic features of a rule that may happen in portrait photography, and it is not explored by the rule lonely.

Figure 4.4 shows some sample poses by an avatar which makes a shot appealing if someone uses it. As observed, these are just for a standing pose, and there are many other combinations as well. Even these poses can be rotated or moved a
little without losing any integrity of the pose. Exploring all of such situations needs a broad rule expansion or decomposition and then clustering to include most combinations with separable borders from each other accurately. Clustering expanded rule-based features based on their importance map gives us different views of a rule, and then we can have a more accurate and robust aesthetic fact model.

A photography rule in an image can be characterized by the reference features of the image. In the process of $K$-order expansion, a rule ($R_i$) produces $K$ number of $R_i$-based features ($\{e_{i,j}^{(k)} : \forall 1 \leq k \leq K\}$) for $i$-th rule $j$-th image, where:

$$\max_{\forall j,k; f_{i,j}, f_{i,k} \in R_i} \{D_{R_i}(f_{i,j}, f_{i,k})\} \leq K.D_{R_i}(e_{i,l}, e_{i,m}); \forall l, m \in 1, 2, ..., K. \quad(4.16)$$

Large $K$ makes the processing time complexity very high. So we should reduce unnecessary features’ dimension using clustering or any unsupervised learning like
PD2-clustering algorithm [129] or K-means algorithm [132], i.e.:

\[
\{e_{i,1}, e_{i,2}, \ldots, e_{i,L}\} = - \arg \max_S \sum_{j=1}^{L} \sum_{e_{i,k} \in s_j} D_{R_i}(e_{i,k}, \mu_{i,j})^p,
\]  

(4.17)

where \(L << K\), and \(S = \{s_1, s_2, \ldots, s_L\}\) is the set of all clusters, and \(\mu_{i,j}\) is the mean value of the cluster \(s_j\). This type of learning can be also done by adding artificial samples to the training dataset. In this approach, we also need to figure out how to leverage and embed the expansion of the rule in our training samples, but the process of the learning is more intuitive.

### 4.3.3.3 Image Parametric Aesthetic Model (iPAM)

The aesthetic features extracted from the image establish the static aesthetic model of the image, but we want to search proximity of an aesthetic feature to select the best neighbor reference feature, and in the next steps, corresponding notion (like an avatar for portrait) of the best reference feature sends as guiding feedback to the photographer. Thus, we need a parametric model in terms of image dynamicity, generalizing the former static model from the aesthetic features.

The parametric aesthetic model of the taken image captures the dynamicity of the image in terms of movable objects such as arms, legs, and cameras. For example, the image horizon or ground is the inertial reference frame of body joints based on Newton’s first law, and respectively the trunk is the inertial reference frame for the other joints. Also, the neck, arms, and legs can move to other locations to some extent and create some other poses.

These joints and their movements can be seen as a dynamic or parametric bag of features in terms of adjustable variables to change the pose and location. The body joints (\(J_i\)) are connected by the body parts, and the parts can be replaced to other positions by a transformation including rotation (\(O\)) or translation (\(T\)). So we have such examples as follows:

\[
J_i^{new} = J_i^{new} + (O_i^{new} \cdot J_i^{old} + T_i^{new}),
\]

(4.18)

where \(J_0\) is the ground or initial inertial reference frame. Now, we can formulate the dynamicity of the end joints by recurring equation 4.18. To find the aesthetic
parametric model of the taken image:

- First, we should find an expanded feature vector of the image in terms of each learned aesthetic fact or rule. In portrait photography, the relative position of the joints with respect to the ground in equation 4.18 can be formulated starting from trunk to end joints of a body where \( T_i \) and \( O_i \) for \( i \)-th joint are our parameters.

- Second, we should make these parametric features constrained to the boundaries of the feasible region of the actions that happen in the image, i.e. \( T_i \) and \( O_i \) are confined to some extent with respect to the kind of the joint. For example, in portrait photography, a forearm cannot bend in opposite direction and the feasible interval is nearly between 0 and 180 degrees. We have:

\[
\forall i, O_{j_i}^{\text{min}} \leq O_i \leq O_{j_i}^{\text{max}}, \quad (4.19)
\]

\[
\forall i, T_{j_i}^{\text{min}} \leq T_i \leq T_{j_i}^{\text{max}}, \quad (4.20)
\]

where \( O_{j_i}^{\text{min}}, O_{j_i}^{\text{max}}, T_{j_i}^{\text{min}}, \) and \( T_{j_i}^{\text{max}} \) are secondary action boundaries of the joints leading to the primary action boundaries 4.4 and 4.5.

Therefore, \( O_i \) and \( T_i \) are our dynamic knobs to give \( J_i \) liveness, as the relation defined in 4.18.

### 4.3.3.4 Image Aesthetics Improvement (iAI)

The image aesthetic improvement (iAI) algorithm searches the neighborhood of aesthetic features to get the optimal reference feature and then maps it to the corresponding movement as feedback to the photographer. It is an optimization framework to get the best state using iPAM and the learned aesthetic model.

Given the distance operator of two aesthetic features explored in step (0) 4.10, we can construct our optimization problem as the minimum distance of the learned aesthetic fact models in Step (1) from the parametric aesthetic model derived in Step (2-a) subject to feasible actions in (2-b). The optimization finds the best pose over all kinds of actions defined in (2-a), and there are limited to the boundaries defined in (2-b).

In case of portrait photography, we define the objective function as a weighted
linear summation of the normalized distances between image parametric aesthetic model and image aesthetic fact model, where the fact model, the weights are known from learning step (1), and the image parametric model is derived from step (2). So, we have:

\[
\{\{O_1, O_2, \ldots\}, \{T_1, T_2, \ldots\}\} = - \arg \max_{O_j, T_j} \sum_{i}^{M} w_i D_{R_i}(R_{i,x}, e_{i,x}(O_j, T_j)),
\]

subject to:

\[
\forall i, j : \theta_{<i-1,i>.,j}^{\text{min}} \leq \theta_{<i-1,i>.,j} \leq \theta_{<i-1,i>.,j}^{\text{max}},
\]

\[
\forall i, j : \phi_{<i-1,i>.,j}^{\text{min}} \leq \phi_{<i-1,i>.,j} \leq \phi_{<i-1,i>.,j}^{\text{max}},
\]

where the \(x\) is the index of the taken photo, \(R_{i,x}\) is the reference feature of \(i\)-th rule and \(x\)-th sample, and \(e_{i,x}(O_j, T_j)\) is the expanded \(R_i\)-based feature of \(x\)-th sample in terms of possible movements \(O_j, T_j\). Note that \(R_{i,x}\) is independent of the movements, and it can be spatial composition reference feature, background aesthetic reference feature, or generic descriptors from CNN [94].

The optimization problem may have (a) solution(s) in the feasible region, or we may have to choose the best aesthetic solution based on other factors such as age, gender, scene type, or minimum activity to change the pose. However, if there is no solution, we can just relax one of the least important aesthetic features and then start over from Step (2) to Step (4), and try to find any solution.

**On-site feedback System:** To illustrate the feedback for the photographer, there are many ideas but one of the practical ideas is using a 2D avatar to animate the transition from the current state to the final optimal state. It is similar to what Lu and Zhang [133] propose 3D and 2D automatic generation of computer animation using a script.

To make it happen, all pose transitions labeled by \(J_i \forall i\) positions are needed in addition to the poses. The animator gets the first and final labels, and calculates all intermediate \(J_i\)s, and then shows the corresponding poses of the \(J_i\)s sequentially from the first frame to the last frame as an avatar animation.
Figure 4.5: The results of CNN (1st row), KNN-SVM (2nd row), and our method (3rd row) for a sample shot at the left side.

### 4.3.4 Performance Accuracy

In this section, we aim to evaluate the performance of the feedback provided by our system and measure how much the solution is close to reality. There is no other similar or comparable system in the literature to compare with our proposed framework. Thus, we run a survey study of various on-site feedback and perform a quantitative evaluation based on the survey results. We usually provide a website for the users’ convenience and run the survey with a tutorial and a bunch of random hypothesis tests.

To evaluate the functionality and the performance of our method and measure how much the recommended photos make sense and are helpful to the photographer, we conduct a quantitative user study based on the human subject results to compare our method with state-of-the-art semantic and scene retrieval based on CNN [114] and KNN-SVM [134]. We select various image queries based on many types of portrait categories such as background scene and semantics, single versus group, full-body, upper-body, facial, standing versus sitting, and male versus female. All 4096 generic descriptors via public CNN model [135] trained on ImageNet [136] are extracted for our large dataset images as well as the features of KNN-SVM-based method [134]. The hypothesis tests are asked through a PHP-based website with usage guidance, and the outputs of the methods are randomly shown in each row to be chosen by more than thirty participants who are graduate students.
Our framework received 65.3% of the 1st ranks among the tests compared to 27.1% CNN as the second rank and 7.6% KNN-SVM as the third rank. Figure 4.5 illustrates the results of all methods (CNN: 1st, KNN-SVM: 2nd, ours: 3rd row) with respect to a similar shot at the left side. As it is realized from Figure 4.5 and our study, the other methods cannot capture portrait categories, scene structure, and corresponding poses of the query shot very well because the badly-posed full-body query shot is suggested as upper-body, facial, and back poses by other category-agnostic methods. As we hierarchically index our dataset by portrait images, portrait categories, and semantic categories, our semantic-aware framework accessing our indexed dataset can retrieve related photography ideas.
Chapter 5

Comprehensive Composition Assistance for Photo Taking

5.1 Introduction

Digital photography is of great interest to many people, regardless of whether they are professionals or amateurs. It has been estimated that over a billion photos are taken every year and they are primarily taken with smartphones. People on social networks often share their photos with their friends. Smartphones’ increasing computing power and ability to connect to more powerful computing platforms via the network make them potentially useful as a composition assistant to amateur photographers. Major smartphone manufacturers have started to introduce on-device photo enhancement capabilities.

Emerging technologies, including artificial intelligence (AI)-chips and AI-aware mobile applications, provide more opportunities for composition assistance. Taking stunning photos often needs expertise and experience at a level that professional photographers have. Like in other visual arts, a lack of a common alphabet similar to music notes or mathematical equations makes transferring knowledge in photography difficult. To many amateurs, as a result, photography is mysterious, and gaining skills is neither easy nor quick. Nonetheless, many people are fascinated by professional-quality photos and desire to have the ability to create similar-quality photos themselves for the scenes or events they are interested in. Because aesthetics in photography is strongly linked to human creativity, it is daunting for an AI to
Figure 5.1: Sample photos retrieved from the dataset based on the photo category and/or subject gender. Each retrieved result shows a collection of photography ideas that can be used by an amateur to compose photos for a given situation.

compose photographs at a given scene or a given studio setup that can impress people in a way professional photographers do. We attempt to connect human creativity as demonstrated through their creative works with AI.

Aesthetics and composition in photography have been heuristically explored as a collection of rules or principles such as balance, geometry, symmetry, the rule of thirds, and framing [34, 37]. It is well known that professional photographers take a large number of pictures, and through their practice, they gain experience and knowledge which in turn enable them to be creative [36]. Some composition rules or principles have been well articulated and many amateurs make use of them in their photo taking. However, the set of known principles can hardly cover the creativity and experience of thousands of photographers around the world. There is no unique photography idea for a given situation, and people have different opinions on those ideas depending on their cultural background, gender, age, experience,
and emotional state. As a result, if the aesthetic quality of photos is quantified by one number, it can only emulate the average opinion of the general public, which may or may not be useful for a particular person.

It would be helpful if an AI can help people explore places [137] and select *photography ideas* from thousands of professional-quality photos for a given scene. The key technical difficulties for accomplishing this goal are (1) finding a suitable mapping between an amateurish photo of a scene and underlying professional-quality photography ideas, (2) handling a virtually unlimited number of photography ideas of a scene, and (3) providing meaningful and intuitive in-situ assistance to the photographer based on personal preference. Using a data-driven approach through recommendations from a large professional-quality dataset, this work tackles these challenges.

The multimedia and computer vision communities have been leveraging some of the photography composition principles for aesthetics assessment [38, 41–44]. Other approaches manipulate the photo [52, 117] to comply with artistic rules, and they are referred to as auto-composition or re-composition. The techniques include smart cropping [53–55, 57–60], warping [61, 62], patch re-arrangement [63–65], cutting and pasting [52, 57], and seam carving [67, 68]. However, they do not help an amateur photographer capture a more impressive photo to begin with. More specifically in portrait photography, there have been rule-based assessment models [69, 70] using known photography basics to evaluate portraits, and facial assessment models [71–75] exploiting features including smile, age, and gender from face. A composition rule-based feedback system [3, 4] has been developed to help amateur photographers by retrieving images with similar known composition rules, but the system is limited to basic composition categories (e.g., horizontal, vertical, diagonal, textured, and centered). The arrangement of dark and light masses, known as “Notan” in visual art, has been used for composition analysis [4]. More recently, perspective-related techniques [5], the triangle technique [6], and portrait composition technique [107] have also been exploited.

We\(^1\) investigate a holistic framework [138, 139] for helping people take a better

\(^{1}\text{This work is based on a paper published in ACM Transactions on Multimedia Computing, Communications, and Applications. Most of the work was done by Farshid Farhat. Mohammad Mahdi Kamani partially did the writing and implementations in section 5.3.3}
shot with regard to their current photography location and need. The framework addresses the differences in preferences of the users by adjusting the ranking process used to retrieve recommendation photos. After getting the first shot from the camera, our framework provides highly-scored related photos as pre-composed “recipes” (i.e. photography ideas) for the user to consider. As an example, regarding personalized criteria (such as photo category and subject gender), Figure 5.1 shows sample results retrieved from the photo dataset. These photos illustrate various locations, scenes, and categories. One can argue that while photos in the same row have the same category or gender, each photo has a photography idea(s) that is different from those used in other photos of the same row. For example, in the 2nd photo from the left in the 1st row, the subject using rule-of-thirds emphasizes the region of interest, and in the 2nd and 3rd photos from the right in the same row, the subjects cross their legs and bend one of the knees to form a triangle in the resulting photo. As mentioned before, the rule-of-thirds and triangle techniques are popular techniques used by professionals. Also, they may use one technique, but the way they use them is different, forming different photography ideas.

To address the complexity of transferring photography idea(s) to a user, we break down the scene that the user wants to take a photo from into composition primitives and then build them up for a better-composed shot using professional-quality photos from the dataset with similar composition primitives. To accommodate the user’s individual preferences, we perform personalized aesthetics-aware image retrieval (PAIR). Figure 5.2 shows the flowchart of our approach for assisting photographers in taking an improved photo. Based on the first query shot, highly-rated photos are retrieved from the collected dataset using the user-specified preferences (USP) and our composition model (CM). Then, the results are shown to the photographer to select some of them as a user-preferred style set. The camera then takes a sequence of shots, from which the one that is the closest match to the style set is chosen. The details of the procedure will be explained later. The main contributions of this work are as follows:

- We propose a new fork-join framework that understands the mapping between a photo and its potential underlying photography idea(s) through decomposing the taken photo into aesthetics/composition-related ingredients (Section 5.3) and followed by composing those ingredients to show recommended ideas to
Figure 5.2: The flowchart of our composition assistance framework: Blue, black, red, and green flows show the user settings, the indexing process, the searching process, and the matching process, respectively. The decomposition box (the box with a dashed line) extracts the aesthetics-related features of the images and computes our composition model. The composition box (another box with a dashed line) recommends some well-composed images in our dataset based on the aesthetics-related features and user-specified preferences.

We design the decomposition step to extract composition primitives of a query shot using various detectors including the newly developed integrated object detector (IOD) (Section 5.3.1), category detector (CaDe) (Section 5.3.2), and our proposed artistic pose detector (ArPose) (Section 5.3.3). The IOD consists of a collection of performance-enhanced detectors including an object detector, a pose estimator, and a scene parser. The integration of them substantially boosts the detection accuracy by the proposed hysteresis fusion
(Section 5.3.1.2). The CaDe has top-down decisive hierarchical clustering (Section 5.3.2.1) and multi-class categorization to leverage genre information (Section 5.3.2.2). The ArPose performs pose clustering (Section 5.3.3.1) to extract pose information using joint to line distance and skeleton context features.

- We address the complexity of transferring photography knowledge, caused by the existence of abundant, diverse, and correlated photography ideas of a scene, by providing personalized meaningful and useful feedback to photographers. We design the composition step to get a similarity score (Section 5.4.1) and perform personalized aesthetics-aware retrieval (Section 5.4.2).

- In our framework, we manage a dataset containing 500K+ photos where 200K+ of them are highly rated covering a large number of photography ideas (Section 5.2) for training and retrieval. Using this dataset, we accommodate users’ needs for composition, by showing a ranked list of photos based on user-specified preferences (USP).

## 5.2 The Dataset

The most valuable resource used by our framework is the collected dataset because it contains a large number of innovative photography ideas from around the world. We have examined photo-sharing websites for photography purposes including Flickr, Photo.net, DPChallenge, Instagram, Pinterest, and Unsplash, but none of them properly cover several categories such as full-body and upper-body in portrait photography as well as landscape photography ideas.

**Portrait and Landscape Dataset:** The dataset is gradually collected by crawling the 500px website which contains photos from millions of photographers around the world who are expanding their social networks of colleagues while exploiting technical and aesthetic skills to make money by marketing their photographs. To get the file list and then the images sorted by rating, we have implemented a distributed multi-IP address, block-free Python script. Nearly half a million images for the current dataset have been collected. The dataset has diverse photography ideas especially for the aforementioned portrait categories (full body, upper body,
facial, group, couple or any two-body, side-view, hand-only, and leg-only) and landscape categories (nature, urban, etc) from highly-rated images taken by mostly photography enthusiasts and professionals. While Figure 5.1 shows sample photos from the dataset, Figure 5.6 in Section 5.6.1 shows the properties of the dataset. As a result, more than 90% of the images were viewed more than 100 times, and nearly half of the images in the dataset had a very high rating between 40 and 50, out of 60.

Automating Dataset Annotation: Our dataset contains 500K+ images where 200K+ of them are highly rated. We have manually annotated around 50K+ of the dataset for training, verification, and testing purposes. More precisely, we annotate 10K+ images for object detector, 5K+ for pose estimator, about 5K for scene parser, and about 25K for portrait categorization. Also, we discard the rest of the training/testing set as they are unrelated. Then, we leverage multiple highly accurate detectors to automate and accelerate the annotation process of the rest of the images. However, the accuracy of our IOD (92.02%) and our CaDe (91.60%) for auto-annotation are high enough to retrieve aesthetics-aware exemplars. Also, the redundancy across our designed detectors makes the annotation process more robust.

5.3 Photo Decomposition

Content-based image retrieval (CBIR) methods help us map unbounded correlated data (e.g. an image) to a bounded range (e.g. feature vector), and then find similar images based on similarity metrics. But there are many restrictions to exploit them directly for applied problems. As mentioned before, there is no limit to innovation in visual arts. Hence, it is very difficult if not impossible for available methods to map an image to a set of useful and related photography ideas which are abundant, diverse, and correlated. Also, as the number of ideas increases, mean average precision (MAP) falls abruptly at the rate of $O\left(\frac{1}{n}\right)$, and manual idea labeling of a large dataset is costly in terms of computational time and available budget.

To recommend better-composed photos to a photographer, we decompose the query image from the camera shot into composition ingredients called aesthetics-aware information. This information includes high-level features (such as semantic
classes, photography categories, human pose classes, subject gender, and photo rating) as well as low-level features (such as color, texture, etc). To accelerate the retrieval process from the dataset based on a query image, we perform the decomposition procedure on all images in the dataset as an offline process, called *indexing*, shown as black arrows in Figure 5.2. We construct the composition model (CM) after indexing the whole dataset. If new images join the dataset, we index them and update our CM. In the *searching* step shown as red arrows in Figure 5.2, we decompose the query image and compare it with our CM. Then, we retrieve the highly-ranked photos from the dataset based on the decomposed values of the query and user-specified preferences (USP).

Through this section, we describe the proposed integrated object detector (IOD) to determine semantic classes in a query image more comprehensively and more accurately than a single object detector. Also, the proposed category detector (CaDe) specifies the photography genre and style. Furthermore, the proposed artistic pose clustering (ArPose) extracts human pose information specifically for portrait photography.

### 5.3.1 Integrated Object Detectors (IOD)

To tackle the problem of classifying a virtually unlimited number of photography ideas, we need composition ingredients of a query shot, and then map them to top-ranked photography ideas. One of these ingredients is semantics inside query shot. To detect these semantics more accurately, we adopted deep-learning architectures and improved the detection accuracy compared to the state-of-art detectors by training our customized architecture on an augmented dataset including common failure cases (CFC) from our dataset, plus other available datasets including MSCOCO [140] and ADE20K [141]. Figure 5.3 illustrates how state-of-the-art object detector (YOLOv3 [108]), human pose estimator (OpenPose [142]), and scene parser (PSPNet [112]) poorly perform on a CFC set of images in our dataset compared to qualitative results from our IOD. Because OpenPose misses at facial photos to detect human parts like the neck in close-up photos and it is not very accurate at “two” or “group” categories to associate parts overlapping. Non-person detection of YOLOv3 under 34% probability is sometimes not reliable, and PSPNet detection is partially not accurate enough at photos with many objects, as it partitions the
photo into small segments and it never considers overlapped area. To improve the accuracy of the detectors, we have changed the deep-learning architecture in terms of reduction layers, transformation parameters such as maximum rotation, crop size, scale min, and max. Because we can speed up the process by changing the reduction layers, and higher rotation and bigger portraits are more important in photography.

We adopted our object detector network architecture inspired by the GoogLeNet model [143] with 24 convolutional layers followed by fully connected layers, but with a simpler reduction to convolution layers to be faster. The output of the network is the bounding boxes of the detected objects with their probabilities. In our model, we do not consider non-person objects whose detection probability is less than 28%, because any wrong detection affects all pixels in the bounding box. As a result, we divide the input image into bigger chunks of $5 \times 5$ grid for higher accuracy and smaller objects which are less important for detection as a secondary subject of photography. Our pose estimator architecture has a multi-stage convolutional neural network with two parallel lines predicting a limb confidence map and encoding the limb-to-limb association inspired by [142]. We adjust the transformation parameters of the architecture including maximum rotation degree to 60, crop size to 500, scale min to 0.6, and scale max to 1.0, since higher rotation degrees and bigger people are used frequently in this work. To design our scene parser, we ignore confusing labels like buildings and skyscrapers. We place the related objects in the same object category. Also, our scene parser architecture exploits a 4-level pyramid pooling module [112] with sizes of $1 \times 1$, $2 \times 2$, $3 \times 3$ and $4 \times 4$ respectively. We do not consider detecting small objects in the scene since they are mostly not the main subject of the photographer.

5.3.1.1 Value Unification

To perform computation on the inference of our customized detectors in the next steps, we need to unify them in terms of pixel-level tensors. We define their scores as $- \log (1 - p)$ for each pixel of the image. The object-ID and its score for each pixel are represented as an $m \times n \times 2$ tensor. Our object detector, scene parser, and pose estimator infer object-IDs respectively across 80 objects, 150 semantics,
Figure 5.3: Qualitative results show the improvement by our integrated object detector (IOD). Each row includes the original, YOLOv3, OpenPose, PSPNet, and the results from our IOD.

and 18 anatomical part IDs. Thus, for each image \((I_{m \times n})\) we have:

\[
T_{I,od}^{I_{m \times n \times 2}} = \begin{bmatrix} t_{i,j,k}^{I_{od}} \end{bmatrix}, t_{i,j,1}^{I_{od}} = C_{i,j}^{I_{id}}, t_{i,j,2}^{I_{od}} = -\log_2 (1 - p_{i,j}^{I_{od}}),
\]

\[
T_{I,sp}^{I_{m \times n \times 2}} = \begin{bmatrix} t_{i,j,k}^{I_{sp}} \end{bmatrix}, t_{i,j,1}^{I_{sp}} = A_{i,j}^{I_{id}}, t_{i,j,2}^{I_{sp}} = -\log_2 (1 - p_{i,j}^{I_{sp}}),
\]

\[
T_{I,pe}^{I_{m \times n \times 2}} = \begin{bmatrix} t_{i,j,k}^{I_{pe}} \end{bmatrix}, t_{i,j,1}^{I_{pe}} = J_{i,j}^{I_{id}}, t_{i,j,2}^{I_{pe}} = -\log_2 (1 - p_{i,j}^{I_{pe}}),
\]

where \(I\) is an input image, \(m\) is the number of rows, \(n\) is the number of columns in the image, \(T_{I,od}^{I}\) is the corresponding tensor of object detector, \(C_{i,j}^{I_{id}} \in \{1, 2, \ldots, 80\}\) is the object-ID of the pixel at \((i, j)\), \(p_{i,j}^{I_{od}}\) is the object-ID probability of the pixel at \((i, j)\), \(T_{I,sp}^{I}\) is the tensor of scene parser, \(A_{i,j}^{I_{id}} \in \{1, 2, \ldots, 150\}\) is the object-ID of the pixel at \((i, j)\), \(p_{i,j}^{I_{sp}}\) is the object-ID probability of the pixel at \((i, j)\), \(T_{I,pe}^{I}\) is the
tensor of pose estimator, \( J^I_{i,j} \in \{1, 2, ..., 18\} \) is the joint-ID of the pixel at \((i, j)\), and \( p^I_{i,j} \) is the joint-ID probability of the pixel at \((i, j)\).

### 5.3.1.2 Hysteresis Fusion

To expand the coverage of the photography idea space in the dataset, all detectable objects in each image using our detectors are leveraged. We maximize the dataset coverage while the accuracy is higher than 90%. Our *hysteresis fusion* optimizes the LOW and HIGH thresholds of the detection probability which is the average probability (or score in Eq. 5.1, 5.2, and 5.3) of the pixels of the object \( X \) for each (object \( X \), detector \( Y \)) binary. If all detectors are below their LOW thresholds for object \( X \), it means there is no object \( X \) in the image. If one of the detectors is above its HIGH threshold, it means there is an object \( X \) in the image. There is a narrow ambiguity region between LOW and HIGH values which covers a few images that we ignore.

To tune the thresholds, we conduct the experiments in Section 5.6.2.4 and consider detection probability as the object detector feature, and normalize area as the pose estimator feature. We get a bi-variate histogram (extendable to N-dimensional histogram for N detectors) in Figure 5.9 illustrating the frequency of the images smart-binned by the normalized object detector and pose estimator scores. Following these thresholds, the accuracy of our IOD scheme is 92.02% (higher than each detector), and 84.7% of the images are covered.

### 5.3.1.3 Object Importance

To prioritize the prominence of the objects in the image, we seek to use the importance map of the objects, because the subject of the image should be more important even if its detection probability is lower. To rank the order of the objects, we exploit the max score multiply by a saliency map \( S \) features with our centric distance \( D \) feature to get our weighted saliency map \( W \).

\[
W^I(i, j) = \max \left( T^I_{od} \cdot T^I_{sp} \right) H \cdot S^I(i, j) \cdot D^I(i, j),
\]

\[
D^I(i, j) = e^{-\|i,j\|^2 / K},
\]
\[ c^I = \frac{\sum_{i,j} S^I(i,j) \cdot [i,j]}{\sum_{i,j} S^I(i,j)} , \]  

(5.6)

where \( W^I(i,j) \) is our weighted saliency map point-wisely for image \( I \), \( \text{max}(.) \) operation is a hysteresis max on the second plane of the tensors (score matrix), \( S^I(i,j) \) is a fast implementation of saliency map of image \( I \) [144], and \( D^I(i,j) \) is our centric distance feature of image \( I \), \( K \) is a tunable constant equal to \( \sum_{i,j} e^{-\|i,j\|_k} \) for image \( I \), the binary value \( c^I \) is the center of the mass coordinate, and \( \|\cdot\|_k \) is the \( k \)-th norm operator where \( k = 1 \) in our experiments. Our weighted saliency map makes the detected objects prioritized, because we sum up the scores from the semantic classes, and we end up with a total score for each semantic class. The output of this step is a weighted vector of detected semantic classes (undetected object has zero weight) in the query image. We show it as the following vector where the elements represent the importance (normalized as a probability) of the corresponding object in the image:

\[
F_{I,iod} = \begin{bmatrix}
  f_{imp}^1 \\
  f_{imp}^2 \\
  \vdots \\
  f_{imp}^{210}
\end{bmatrix},
\]

(5.7)

\[
f_{imp}^k = \frac{\sum_{\forall\text{pixel}(i,j) \in \text{obj}(k)} W(i,j)}{\sum_{\forall i,j} W(i,j)},
\]

(5.8)

where \( \forall k \in \{1,\ldots,210\} \), \( f_{imp}^k \) is the importance value of \( k \)-th object which is the summation of the weighted saliency of its every pixel \((i,j)\), and \( F_{I,iod} \) is the importance vector of all objects.

### 5.3.2 Category Detector (CaDe)

The photo categories in portrait include full-body, upper-body, facial, side-view, two (couple or two people), group (more than two people), faceless, headless, hand-only, and leg-only, which are ten classes. In landscape photography, there are sea, mountain, forest, cloud, and urban, which are five classes. While we focus on portrait and landscape photography genres, we believe that this work can be extended to other genres as well. Knowing the photo genres and categories helps our framework guide the photographer more adequately because it retrieves better-related results based on the photographer’s preferences. The downside can be the low coverage or a limited number of contents on the leaves of this hierarchical
tree of the photo styles, but our comprehensive dataset addresses this potential issue.

5.3.2.1 Top-down Hierarchical Clustering

To distinguish a portrait from a landscape photo, the number of people in the image is estimated by the max (union) number of person-IDs higher than their corresponding HIGH thresholds across the detectors in integrated object detector (IOD). If the score for detecting a person is lower than a LOW threshold for all detectors in IOD (intersection), there is no person in the image. Then, if there is a water-like, mountain-like, plant-like, cloud-like, or building-like object in the image with a total area higher than 26.5% (empirically tuned for landscape), the landscape category will be recognized as well. Otherwise, the image is ignored because its subject is not for portrait or landscape photos.

5.3.2.2 Portrait Multi-class Categorization

To automate an efficient and accurate portrait categorization, we formulate the problem as a multi-class error-correcting output code model using multiple support vector machine binary learners (let say ECOC-SVM). The inputs are our feature vector and the corresponding class labels. Since we are using 10 portrait categories or unique class labels, it needs 55 (= 10 × (10 + 1)/2) binary SVM learners with radial basis function (RBF or Gaussian) and a one-vs-one coding design. We have annotated 5% (about 25K+) of portrait photos uniformly selected at random from the dataset as the ground truth of the portrait categories. Then, we train an ECOC-SVM with the feature vectors and the corresponding labels of 80% (about 20K) of our ground truth and leave the rest for testing our ECOC-SVM. Our feature vector for each photo includes 40 different features as follows:

- General MAX: (1,2) max scores for detected people, and (3,4) max areas for the detected people from object detector and pose estimator.

- Intersection Area: (5) the area(s) of the people with the highest detection probability, (6,7) the scores of these people, (8,9) the areas of these people from object detector and pose estimator.

- Number of people: (10,11) number of people higher than the HIGH threshold
for each detector, (12,13) number of people with area higher than 5% for each
detector from object detector and pose estimator, (14) max of feature # 10
and feature # 11, (15) max of feature # 12 and feature # 13, (16) max of
feature # 14 and feature # 15.

- Limb Features: (from 17 to 40) the limbs respectively including nose, neck,
right shoulder, right elbow, right wrist, right hand, left shoulder, left elbow,
left wrist, left hand, right hip, right knee, right ankle, right leg, left hip, left
knee, left ankle, left leg, right eye, left eye, eyes, right ear, left ear, ears which
add up to 40 features.

The output of this step for an image query is the following unitary vector that
shows its category:

\[
F_{I,\text{cade}} = \left[ f_{1}^{\text{facial}} f_{2}^{\text{fullbody}} f_{3}^{\text{upperbody}} f_{4}^{\text{two}} f_{5}^{\text{group}} f_{6}^{\text{sideview}} f_{7}^{\text{leg}} f_{8}^{\text{noface}} f_{9}^{\text{hand}} f_{10}^{\text{nohead}} \right], \quad (5.9)
\]

where \( F_{I,\text{cade}} \) shows the unitary category vector of the image \( I \) by CaDe detector,
and only one of the vector elements is one and the rest are zero. The mean average
accuracy of our category detection is shown in Table 5.4 in Section 5.6.2.5 for the
dataset images divided by various styles.

5.3.3 Artistic Pose Clustering (ArPose)

Posing, one of the essential ingredients of portrait photography, could substantially
differentiate between amateur and professional shots. Having little experience in
portrait photography, finding correct postures, or coming up with novel poses
is hard for amateur photographers. Hence, it is vital for our system to have an
understanding of different poses and how to categorize them.

Although RTMPPE extracts body joints in images, these joints are merely
considered as our features for pose detection. We use two sets of features on top
of joint coordinates to define the distance between different poses. These sets
of features are scale-invariant, thus regardless of the scale of the human body in
images, we measure the similarity of two poses. These features are defined as follows:
• Joint to Line Distance ($\text{dist}_{\text{JtoL}}$): This distance vector consists of the distances between each joint and any line that connects two other joints. To have a scale-invariant distance, we normalize the distances with the maximum distance of each body in the image. Having the joint $j_l$ and the line crossing two other joints, $j_m$ and $j_n$, the $\text{dist}_{\text{JtoL}}$ is calculated as follows:

$$\text{dist}_{\text{JtoL}}(l,m,n) = 2S_{\Delta lmn}/||j_m - j_n||_2,$$

(5.10)

where $S_{\Delta lmn}$ is the area under the triangle formed by three joints. Based on the total number of joints in each body, which is 18, and the total number of different distances is $18 \times \binom{17}{2} = 2448$.

• Skeleton Context (SC): Also, another scale-invariant feature vector from previous work [125, 126] is leveraged. Skeleton context is a polar histogram of each point in the skeleton indicating the angular and distance distribution of other points in the skeleton around that point. We benefit from the angular distribution of each point and create an $18 \times 18$ angular matrix for each body in the image. These features are designed to capture the relative position of each joint with respect to other points, hence, they are used as a measure of distance between different poses. We concatenate these features together to use the relative distance and polar information of both. Next, we use these features to cluster images based on various poses.

5.3.3.1 Pose Clustering

To rank the professional poses and find similar ones to the pose in the query image, we use clustering methods that distinguish body postures and group similar ones using the features explained in Section 5.3.3. To do so, we use clustering algorithms, including k-means and Deep Embedding [145], and compare the results of these clustering methods. To determine the optimal number of clusters for the dataset, there are several heuristic methods including but not limited to elbow [146] and silhouette [147] methods. Having too many clusters would diminish the novelty and diversity of the results, in the sense that it tries to have samples as close as possible to one cluster. On the other hand, keeping the number of clusters low
Figure 5.4: Qualitative results of major clusters derived from our algorithm on the portrait dataset. Each row represents the top poses of each cluster.

would affect the quality of clustering, such that irrelevant poses might appear in the same cluster. The result of our experiment using the elbow method shows that
the optimal number of cluster heads is around 12-15 as depicted in Figure 5.10 in Section 5.6.2.6.

Then, we set up two clustering algorithms, k-means and Deep Embedding Clustering (DEC). For k-means, the only adjustable parameter is the number of clusters, but for DEC, we should set up the autoencoder network in addition to the number of clusters. As suggested by Xie et al. [145] and tested by ourselves, the network with 4 layers of encoder consisting of 500, 500, 2000, and 10 neurons in each unit performs astonishingly well on the clustering task of different supervised datasets including but not limited to MNIST [148], STL [149], and REUTERS [150]. Although DEC works great on these supervised datasets, it has not been tested on an actual unsupervised dataset, simply because there is not a gold standard to evaluate the performance on those datasets. However, visual data like the portrait dataset reveals how these algorithms perform, based on human-eye evaluation of the results. Hence, we compare the results of this deep model for clustering with our feature-based k-means clustering. In k-means, to define the probability that
each sample is in the cluster or the degree to which each sample belongs to a cluster, we use the same quantity in fuzzy C-means clustering [151]:

\[ q_{ij}^{-1} = \sum_{k=1}^{K} \left( \frac{||x_i - c_j||^2}{||x_i - c_k||^2} \right)^{\frac{2}{m-1}}, \]  

(5.11)

where \( x_i \) is the sample, \( c_j \) is the center of the cluster \( j \), \( m \) is a positive real number greater than 1 which defines the smoothness of the function, and \( q_{ij} \) represents the probability that the sample belongs to the cluster. Also, DEC has defined a similar quantity [145] using Student’s t-distribution:

\[ q_{ij} = \frac{(1 + ||z_i - c_j||^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1 + ||z_i - c_{j'}||^2/\alpha)^{-\frac{\alpha+1}{2}}}, \]  

(5.12)

in which \( z_i \) is the embedded version of \( x_i \), and \( \alpha \) is the degree of freedom in Student’s t-distribution. Using these metrics we estimate the probability that each sample belongs to a cluster. Hence the output feature of the ArPose module would be in the form of:

\[ F_{I,\text{arpose}} = [f_{I,1}^{\text{arpose}} \ f_{I,2}^{\text{arpose}} \ \cdots \ f_{I,K}^{\text{arpose}}]^T, \]  

(5.13)

where \( f_{I,j}^{\text{arpose}} = q_{I,j} \) is the probability that the pose detected in image \( I \) belongs to the \( j^{th} \) cluster from our pool of \( K \) clusters as defined above.

The qualitative results of the k-means-based clustering algorithm in Figure 5.4 show the top-ranked poses in major clusters. Those of the DEC algorithm are in Figure 5.5. The images are ranked based on their probability computed in Eq. 5.11 and 5.12. As shown in the figures, k-means clusters surprisingly better than DEC, that is, different clusters distinguish different pose styles and each cluster represents a visually almost similar pose. However, DEC fails to accomplish the goal of the clustering task based on human poses. Since the input features are intelligently chosen to be related to the goal, the input space is linearly separable, however, the result of the DEC shows information loss in the autoencoder. We tried the k-means algorithm with PCA to reduce the dimension of the input space to 10 (as it is in the output of the autoencoder in DEC), and still the results of the k-means surpass DEC’s. Through that, we successfully cluster the portrait image
and retrieve almost similar poses or novel ideas in that pose cluster based on the probability of the poses.

The other properties such as rating, tags, and gender in the shot are also extracted from the image and its descriptor. For the low-level features, we collect all 4096 generic descriptors via a public pre-trained CNN model [135] on ImageNet [136] and the conventional features of Mitro’s method [134] as shown in the following equation. Note that there is no limit to collect any other aesthetics-aware information from query images to extend this work depending on photo genre.

\[ F_{I,\text{vgg}} = \begin{bmatrix} f_{I,1}^{\text{vgg}} & f_{I,2}^{\text{vgg}} & \ldots & f_{I,4096}^{\text{vgg}} \end{bmatrix}^T, \]

where \( F_{I,\text{vgg}} \) is a vector containing generic features of image \( I \), and \( f_{I,i}^{\text{vgg}} \) is \( i \)-th generic feature. The superscript “\( T \)” represents the transpose of the vector/matrix. Also, we extract available statistical data via the image properties including rating, view counts, and gender. Then, we have them as follows:

\[ F_{I,\text{stat}} = \begin{bmatrix} f_{I,1}^{\text{rating}} & f_{I,2}^{\text{views}} \end{bmatrix}, \]

\[ F_{I,\text{gender}} = \begin{bmatrix} f_{I,1}^{\text{male}} & f_{I,2}^{\text{female}} & f_{I,3}^{\text{unknown}} \end{bmatrix}, \]

where \( F_{I,\text{stat}} \) is a vector containing the statistical data of image \( I \) including its rating \( f_{I,1}^{\text{rating}} \) and its view counts \( f_{I,2}^{\text{views}} \). Furthermore, \( F_{I,\text{gender}} \) is a vector containing the gender specification of image \( I \) represented by \( [1 \ 0 \ 0] \) as male, \( [0 \ 1 \ 0] \) as female, or \( [0 \ 0 \ 1] \) as unknown.

### 5.3.4 Construction of Composition Model

To aesthetically index all photos in our dataset and easily search by composition features, we decompose them into the following feature vectors and construct our composition model (CM). In fact, \( F_{I,\text{vgg}}, F_{I,\text{iod}}, F_{I,\text{cade}}, F_{I,\text{apose}} \) and other aesthetics-aware information for all \((\forall i)\) images are extracted and appended to corresponding matrices respectively including deep-learned generic features \( M_{\text{vgg}} \) (for color, texture, and edges), all detected objects \( M_{\text{iod}} \), photography category \( M_{\text{cade}} \), artistic pose \( M_{\text{apose}} \), statistical features \( M_{\text{stat}} \) and detected gender \( M_{\text{gnd}} \). Algorithm 2 describes different steps of our decomposition method precisely to make each row.
of our composition model. We have:

\[ F_{i} = [F_{i,\text{vgg}} \ F_{i,\text{iod}} \ F_{i,\text{cade}} \ F_{i,\text{ap}} \ F_{i,\text{stat}} \ F_{i,\text{gnd}}] , \]  

(5.17)

where \( i \in \{1, \ldots, N\} \), \( F_{i} \) is the feature vector of the image \( I_{i} \). Then, we compute the corresponding feature matrix.

\[ M^{T}_{\text{feat}} = \begin{bmatrix} F_{T,\text{feat}}^{T} & F_{T,\text{feat}}^{T} & \cdots & F_{T,N,\text{feat}}^{T} \end{bmatrix} , \]  

(5.18)

\[ M = [M_{\text{vgg}} \ M_{\text{iod}} \ M_{\text{cade}} \ M_{\text{ap}} \ M_{\text{stat}} \ M_{\text{gnd}}] , \]  

(5.19)

\[ M^{T} = \begin{bmatrix} F_{T,1}^{T} & F_{T,2}^{T} & \cdots & F_{T,N}^{T} \end{bmatrix} , \]  

(5.20)

where \((\cdot)^{T}\) is transpose operation, “feat” is feature type from the set \{vgg, iod, cade, ap, stat, gnd\}, matrix \( M_{\text{feat}} \) is the corresponding feature matrix containing feature vector of each image in each row. The final feature matrix \( M \) is the composition model matrix which is the concatenation of all feature matrices or equivalently all feature vectors.

### 5.4 Composition of Visual Elements

Image retrieval methods want to optimize [152, 153] or customize [154] the process of retrieving images with similar semantics in specified regions, which is not an image aesthetics nor composition-related procedure. For example, an amateurish image may focus on a less important semantic as a photo subject or assign a region of interest to a less important semantic, but the retrieval results from our professional-quality dataset are free of such mistakes.

The goal of our composition step is to gather all composition elements from the previous step and recommend related photography ideas from our collected dataset satisfying personal preferences and aesthetics-aware information of the query image. The input to this step is the decomposed values of the image query and user-specified preferences (USP) with our composition model (CM). The output is a collection of well-composed images from the dataset recommended to the user. For example, if we focus on portraits, we desire feedback that contains well-posed portraits with similar semantics and category but better composition.
Algorithm 2: Decomposition

1. **Input:** image query $Q$.
2. **Output:** the feature vector of the query $F_Q^T$.

```plaintext
1: procedure Decomposition($Q$)
2:   Get the generic features of the query:
3:     $F_{Q, vgg} \leftarrow \begin{bmatrix} f_{Q,1}^{vgg} & f_{Q,2}^{vgg} & \cdots & f_{Q,4096}^{vgg} \end{bmatrix}^T$.
4:   Get stat and gender features:
5:     $F_{Q, stat} \leftarrow \begin{bmatrix} f_{Q,1}^{rating} & f_{Q,2}^{views} \end{bmatrix}^T$.
6:     $F_{Q, gnd} \leftarrow \begin{bmatrix} f_{Q,1}^{male} & f_{Q,2}^{female} & f_{Q,3}^{unknown} \end{bmatrix}^T$.
7:   Get the tensor of the object detector (od):
8:     $T_{Q, od}^{m \times n \times 2} \leftarrow Object\_Detect(Q)$.
9:   Get the tensor of the scene parser (sp):
10:    $T_{Q, sp}^{m \times n \times 2} \leftarrow Scene\_Parse(Q)$.
11:  Get the tensor of the pose estimator (pe):
12:    $T_{Q, pe}^{m \times n \times 2} \leftarrow Pose\_Estimate(Q)$.
13:  Get the center of the mass coordinate of the query:
14:    $c_Q \leftarrow \sum_{i,j} \frac{\|\left[i,j\right]-\left[0,0\right]\|_k \times S(i,j)}{\sum_{i,j} \|\left[i,j\right]-\left[0,0\right]\|_k}$.
15:  Get the centric distance feature of the query:
16:    $D^{Q}(i,j) \leftarrow \frac{1}{K} e^{-\frac{\|\left[i,j\right]-c^Q\|_k}{K}}$.
17:  Get the weighted saliency ID map for the query:
18:    $C^{Q}(i,j) \leftarrow S^{Q}(i,j) D^{Q}(i,j)$.
19:    $W^{Q}(i,j) \leftarrow \max\left(T_{Q, od}^{*} , T_{Q, sp}^{*}\right) C^{Q}(i,j)$.
20:  Get the IOD feature vector as Eq. 5.8:
21:    $f_k^{imp} \leftarrow \frac{\sum_{\forall\left[i,j\right] = k} W^{(i,j)}}{\sum_{\forall\left[i,j\right]} W^{(i,j)}}$, $\forall k \in \{1,2,\cdots,210\}$.
22:    $F_{Q, iod} \leftarrow \begin{bmatrix} f_{1}^{imp} & f_{2}^{imp} & \cdots & f_{210}^{imp} \end{bmatrix}$.
23:  Get the category feature $F_{Q, cade}$ as Eq. 5.9.
24:  Get the artistic pose feature $F_{Q, ap}$ as Eq. 5.10.
25:  Get the whole feature vector:
26:    $F_{Q}^{T} = \begin{bmatrix} F_{Q, vgg}^{T} & F_{Q, iod}^{T} & F_{Q, cade}^{T} & F_{Q, ap}^{T} & F_{Q, stat}^{T} & F_{Q, gnd}^{T} \end{bmatrix}$.
27:  If $Q$ is a dataset image, add $F_{Q}^{T}$ to the last row of composition model matrix $M$ in Equation 5.20, otherwise the output is used in Algorithm 3 to retrieve better composed photos and match with the final shot.
28: end procedure
```

As we have collected a dataset containing generally well-composed images, we should dig into the dataset and look for images with “pretty” similar color, pattern, category, pose, or object constellation where the term “pretty” is framed by USP
to address the user’s needs and subjectivity. The existence of this professional-quality dataset makes it possible that the retrieved photos have highly accepted photography ideas by the people. Our image retrieval system is not supposed to find images with exactly similar colors, patterns, or poses, but it finds images with a better composition having similar semantic classes, category, or pose. Thus, the location of the movable objects does not matter, but the detected objects are important.

5.4.1 Similarity Scores and Normalization

Having our composition model for all images in our dataset and the query image, we first calculate the similarity score between the query image and any image in the dataset across the detectors. The similarity metric of generic VGG features (5.14) is the multiplication of the matrix $M_{\text{vgg}}$ by the query vector $F_{\text{vgg}}$. Similarly, the category detector has a matrix by vector multiplication. For integrated object detectors, we use the Gaussian function after masking unrelated objects. For statistics and gender information, it is formulated as follows:

$$S_{\text{vgg}}(I, Q) = F_{I, \text{vgg}}^T F_{Q, \text{vgg}},$$

(5.21)

$$S_{\text{cade}}(I, Q) = F_{I, \text{cade}}^T F_{Q, \text{cade}},$$

(5.22)

$$S_{\text{iod}}(I, Q) = e^{-(\sum(F_{I, \text{iod}} \circ \text{sign}(F_{Q, \text{iod}}) - F_{I, \text{iod}}))^2},$$

(5.23)

$$S_{\text{stat}}(I, Q) = |f_{I, 1}^{\text{rating}} - f_{Q, 1}^{\text{rating}}|,$$

(5.24)

$$S_{\text{gender}}(I, Q) = \begin{cases} 1, & \text{if } F_{I, \text{gender}} = F_{Q, \text{gender}} \\ -1, & \text{otherwise} \end{cases},$$

(5.25)

where $F^T$ means the transpose of $F$, $e$ is a mathematical constant about 2.72, the $\circ$ operation is the element-wise multiplication, $\text{sign}(.)$ is the sign function operating on each element separately. Also, $S_{\text{vgg}}(I, Q)$, $S_{\text{cade}}(I, Q)$, $S_{\text{iod}}(I, Q)$, and $S_{\text{stat}}(I, Q)$ are similarity score values between image $I$ and image $Q$ respectively for generic CNN descriptors, category detection, integrated object detectors, and statistics and gender information. The similarity score function is easily generalized to a function between two different sets of images, $i.e.$, $I_{m \times 1}$ and $Q_{n \times 1}$ can be a set of images not only one image, and the output will be an $m \times n$ matrix. Since we want to score the similarity between the images in our dataset (say $I$) and an image query...
(Q), in the above equations, vector $F_{I, \text{det}}^{t}$ will be substituted by matrix $M_{\text{det}}$, and
the output will be a similarity vector, while “det” can be any detector:

$$\text{det} \in \{\text{vgg, iod, cade, arpose, stat, gender}\}. \quad (5.26)$$

To make the scores uniform across various detectors, we normalize each detector score vector by dividing by the summation of the whole output. Thus, each detector’s similarity score is like a probability distribution over all images. We have:

$$S^{N\text{feat}}_\text{I,Q} = \frac{S_{\text{feat}}(\text{I,Q})}{\sum_{i \in \text{I}, q \in \text{Q}} S_{\text{feat}}(i,q)}, \quad (5.27)$$

where $S^{N\text{feat}}_\text{I,Q}$ is a normalized similarity score matrix between each image in $\text{I}$ and each image in $\text{Q}$ for detector $\text{feat} \in \{\text{vgg, iod, cade, arpose, stat, gender}\}$. Also, we combine the similarity scores across various detectors to create a tensor of similarity scores for each pair of images from $(\text{I}, \text{Q})$. We have:

$$S^{N}_\text{I,Q} = \begin{bmatrix}
S^{N}_{\text{vgg}} & S^{N}_{\text{iod}} & S^{N}_{\text{cade}} & S^{N}_{\text{arpose}} & S^{N}_{\text{stat}} & S^{N}_{\text{gender}}
\end{bmatrix}, \quad (5.28)$$

where $S^{N}_\text{I,Q}$ is a tensor of size $d \times m \times n$ where $d$ is the number of detectors (||feat|| here is 6), $m$ is the number of images in $\text{I}$, and $n$ is the number of images in $\text{Q}$.

### 5.4.2 User Preferences and Ranking

Prior works have explored feature-based, example-based, and list-based personalized ranking systems for amateur photographs using conventional aesthetic qualities and personal preferences [155]. The example-based and list-based approaches are non-scalable when we have a large dataset as is in our approach. To rank the exemplar list, we multiply user-specified preferences as a probability vector containing the importance weights with our compositional primitive feature matrix. Adjusting weight vector can also cover decision tree-based ranking approaches as well. Now, we have:

$$W_{\text{USP}} = [W_{\text{vgg}} \ W_{\text{iod}} \ W_{\text{cade}} \ W_{\text{arpose}} \ W_{\text{stat}} \ W_{\text{gender}}]^T, \quad (5.29)$$
where \( W_{USP} \) is a \( d \times 1 \) vector showing the weights of the user for each detector, and “T” shows the transpose operation. Then, to retrieve the highest-ranked candidates as the results, the normalized similarity score matrix is multiplied by the USP vector. Consequently, we have:

\[
V_{\text{pref}}(I, Q) = W_{USP}^T S^N(I, Q),
\]

where \( V_{\text{pref}}(X, Q) \) is the user’s preferred image vector, and if we find the \( K \) top-ranked entries with respect to the vector values, the indexes of these entries represent the best high-quality recommendations to the image query \( (Q) \). The results for some queries with USP are shown in Figure 5.11 separated by the query image, the baseline retrieval with equal weights, and the retrieval with proper USP.

## 5.5 Matching

Professional photographers arrange the photograph from top to down (general to detail) step-by-step, while there are many to-do lists and not-to-do lists for photography in his/her mind. We want to create the same environment in a smart camera to accompany an amateur photographer gradually to his/her perfect shot. From the composition step, we retrieve the proper photography ideas given a query shot from the camera. We assume that the photographer has chosen some of the retrieved images as a preferred style set, and disregarded the others as an ignored set. Now we explain how to capture the proper moment of the subject in the scene, and trigger the “moment shot” for the camera.

### 5.5.1 Pose Shot

The best-fitting genre for the matching step is the portrait photography that we start with, and then we extend to the general genre. The variant component in our framework is the human pose. In this scenario, it is assumed that the user has no personal preference on a human pose, \( i.e. \), the user has given zero weight to the ArPose detector to see various proper poses via PAIR, and then the user has selected a preferred style set from the available choices. In this case, we need a mechanism to continue guiding the user to a desired shot by tracking his/her pose via a camera viewfinder.
The relative positions of the human body components (including nose, eyes, ears, neck, shoulders, elbows, wrists, hips, knees, and ankles) with respect to the nose position as portrait origin are consisting our pose model. Preferably, we would like to start from the position of the nose \((J_0 = (0, 0))\) that is connected to neck \((J_1)\), right eye \((J_2)\), and left eye \((J_3)\) are connected to right ear \((J_4)\) and left ear \((J_5)\) as they are on a plane of the head. Also, shoulders \((J_6\) and \(J_7)\) are recognized by a length and an angle from neck, and similarly elbows \((J_8\) and \(J_9)\) from shoulders, wrists \((J_{10}\) and \(J_{11})\) from elbows, hips \((J_{12}\) and \(J_{13})\) from neck, knees \((J_{14}\) and \(J_{15})\) from hips, and ankles \((J_{16}\) and \(J_{17})\) from knees, i.e. these joints are connected as follows:

\[
\text{Pre} (J_i) = J_0, \ i \in \{0, 1, 2, 3\}, \quad (5.31)
\]
\[
\text{Pre} (J_j) = J_1, \ j \in \{6, 7, 12, 13\}, \quad (5.32)
\]
\[
\text{Pre} (J_k) = J_{k-2}, \quad (5.33)
\]
\[
k \in \{4, 5, 8, 9, 10, 11, 14, 15, 16, 17\}.
\]

Thus, we always calculate the absolute position using 2D polar coordinates as follows:

\[
J_i = J_j + r_{i,j} e^{i\theta_{i,j}}, \ i \in \{0, 1, 2, ..., 17\}, \quad (5.34)
\]

where \(j = \text{Pre}(i)\) i.e. part \(j\) is the previous part connected to part \(i\), \(r_{i,j}\) is the length from joint \(J_i\) to joint \(J_j\), \(\theta_{i,j}\) is the angle between the line from joint \(J_i\) to joint \(J_j\) and the line from joint \(J_j\) to joint \(\text{Pre}(J_j)\), and the line crossing \(J_0\) is the image horizon. \(i\) is the unit imaginary number. Note that for a 2D human body \(\{r_{i,j} \mid \forall i, j\}\) are fixed, but \(\theta_{i,j}; \forall i, j\) can be changed to some fixed not arbitrary extents. Similarly, having 3D pose-annotated/estimated single depth images, we can calculate the relative 3D position of the joints using spherical coordinates.

Thus, we have such action boundaries for joints as follows:

\[
\theta_{i,j}^{\text{min}} \leq \theta_{i,j} \leq \theta_{i,j}^{\text{max}}, \ j = \text{Pre}(i), \quad (5.35)
\]
\[
\phi_{i,j}^{\text{min}} \leq \phi_{i,j} \leq \phi_{i,j}^{\text{max}}, \ j = \text{Pre}(i). \quad (5.36)
\]

As a result, a human body pose \((J)\) is represented by:

\[
J^k = (J^k_1, J^k_2, ..., J^k_{17}) , \quad (5.37)
\]
where $J^k$ is the pose for $k$-th person (or $k$-th image with one person), and $\forall i \in \{1..17\} : J^k_i$ is the $i$-th coordinate of the $k$-th person. Also, we need a distance metric to calculate the difference between two pose features. Thus, we define the distance metric as follows:

$$D(J^k, J^l) = \sum_{i=1}^{17} \| \text{Phase}(J^k_i) - \text{Phase}(J^l_i) \|_q ,$$

(5.38)

where $D(.)$ is the distance operator, $J^k$ is the pose feature for $k$-th person (or $k$-th image with one person), $\forall i \in \{1..17\} : J^k_i$ is the $i$-th coordinate of the $k$-th person, and $\| . \|_q$ (usually L1-norm or L2-norm) is the $L_q$-norm function of two equal-length tuples. Our chosen function to track phase, $\text{Phase}(Ji)$, is $\sin(\theta_{i,i-1})$ which $\theta_{i,i-1}$ (the angle between current joint and the previous one) is from $-\pi/2$ to $+\pi/2$. Because the length of the limb may change by going far or near, but the angles between consecutive limbs matter for posing purposes.

Now, the camera may take and hold several photos gradually from the scene, and finally choose the best among them to save onto the camera storage. Our matching algorithm searches among the taken photos to get the nearest pose to one of the collected ideas. The problem is formulated as an integer programming problem to find the best seed among all photography ideas. Given the distance operator of two pose features explored in 5.38, we construct our optimization problem by maximizing the difference of the minimum distance of the ignored set and the minimum distance of the preferred style set of taken photos. Mathematically, we compute the following optimization problem subject to 4.4 and 4.5:

$$I_w = \arg \max_{\forall I_i \in I^t} \left( \min_{\forall Q^g_j \in Q^g} D(J_i^g, J^l_i) - \min_{\forall Q^d_k \in Q^d} D(J_i^d, J^l_i) \right) ,$$

where $I_w$ is the wish-image, $I^t$ is the set of taken photos, $Q^g$ is the set of ignored ideas, $Q^d$ is the set of preferred ideas, $D(.)$ is the distance operator in 5.38, and $J^x$ is the pose for $x$-th image with one person in 4.6. The optimization problem in continuous mode (not over all taken image set) may have (a) solution(s) in feasible region, and in the L1-norm case, it is equivalent to multiple linear programming problems but the complexity of the problem is exponential. Further, the solution does not always give the desired shot.
5.5.2 User Favorite Shot

Given a query shot from the camera, related photography ideas have been already retrieved. Suppose that the photographer selects a preferred style set, denoted as $\mathbb{C} = \{C_1, C_2, ..., C_m\}$, and we also have a set of shots from camera called next query shots, denoted as $\mathbb{Q} = \{Q_1, Q_2, ..., Q_n\}$. The problem of finding the user favorite shot among query shots while satisfying the closest similarity score to the preferred style set is integer programming. We have:

$$Q_{fav} = \arg \max_{q \in \mathbb{Q}} \sum_{j \in \{1, ..., m\}} W^T_{USP} S^N(C_j, q),$$

(5.39)

where $Q_{fav}$ is the favorite shot, $\mathbb{Q}$ is the set of the query shots by camera, $W^T_{USP}$ is the transpose of the user-specified preference vector, and $S^N(C_j, q)$ is the similarity vector between query $q$ and each photo $C_j$ in the preferred style set of the user $\mathbb{C}$. The computational detail of the composition and matching steps has been explained in Algorithm 3.

Mathematically $\mathbb{Q}$ set is not finite, or its size $n$ is not bounded. Also, there are many constraints such as color value ranges, human pose angles, category limits, etc. In the reality, the number of query shots is limited, and the matching solver gradually determines and updates the user’s favorite shot. But the solution is not necessarily optimal, because finding the optimal shot needs the whole shot space which is impractical. The good news is that the user can follow the retrieved professional shots to optimize his/her photography adventure, and the last shot would be close enough to the optimal shot.

Our approach to giving hints to the user includes two steps: 1) defining the query shot space with dynamic parameters in the scene like movable objects or human pose, 2) finding the max over the defined space. This second step is similar to the pose shot approach, and some extra parameters such as photographic lighting may be adjustable as well. The solution of the problem can give a hint to the user to make a change in his/her lighting condition, pose, or any other dynamic parameter.
Algorithm 3: Composition and Matching

1. **Input:** query $Q$, user preference $W_{USP}$, and the set of the images in the 500px dataset $I$.

2. **Output:** user favorite shot $Q_{fav}$.

   1. procedure $IdeaRetrieval(Q, W_{USP}, I)$
   2. Get $F_Q^I$ from Algorithm 2.
   3. Get the similarity score through Eq. 5.25, 5.27, and 5.28:
      \[
      S^N(I, Q) = [S^N_{vgg} S^N_{iod} S^N_{cad} S^N_{arpose} S^N_{stat} S^N_{gender}]
      \]
   4. Get the preferred image vector through Eq. 5.30:
      \[
      V_{pref}(I, Q) = W_{USP} S^N(I, Q)
      \]
   5. Retrieved_Indexes ← Index_Sort($V_{pref}(I, Q)$)
   6. Show_Top(Retrieved_Indexes)
   7. Now, the user selects some of the retrieved results, and the camera takes multiple shot as $Q$.
   8. Find the favorite shot through Eq. 5.39:
      \[
      Q_{fav} = \arg\max_{q \in Q} (\sum_{j \in 1,2,\ldots,m} (W_{USP}^t S^N(C, q)))
      \]
   9. Take $Q_{fav}$ as user favorite shot.
   10. end procedure

5.6 Experiments

In the following sub-sections, we describe our experimental results which are categorized into different components of our method including (i) the dataset, (ii) the decomposition step, and (iii) the composition step. Furthermore, the decomposition step has multiple parts to demonstrate the effectiveness of our method compared to an available state-of-the-art or our baselines.

5.6.1 Dataset Properties

We have collected the images in the portrait and landscape categories from 500px Website and saved them as smaller images where their highest dimension has been resized to 500 pixels. Then, we have collected available metadata for each image including the number of views, the average ratings, the number of vote clicks, and the number of favorite clicks. We conduct statistical experiments to get the properties of the collected dataset. Because some of these properties change dramatically in linear scale, their trends are captured intuitively better in the logarithmic x-axis. Figure 5.6 illustrates the distributions of the view counts,
ratings, vote counts, and favorite counts of the dataset. Each bar represents a bin where its interval is from the corresponding number written under the bin to right before the number written under the next bin. Figure 5.6 shows that most of the images have been seen more than 100 times, i.e., 500px Website has a live community while many images have at least 1-10 votes or favorite clicks. Having a rating higher than 10 is considered high because the rating trend changes its slope direction from bin 0-9 to bin 10-19 negatively, and after that, the slope will positively grow until bin 40-49. Most of the images in the dataset have a rating of more than 40 which is a very high rating, and it indicates that the 500px community of the photographer has many highly-rated photos.

5.6.2 Decomposition Analysis

To show the effectiveness of our decomposition step, we conduct the following experiments on the object detector, the human pose estimator, and the scene parser used in our framework. Also, we examine the hysteresis fusion, the category detector, and the pose clustering.

5.6.2.1 Object Detection

Our object detector network contains 24 convolutional layers with two fully connected layers (mentioned in Section 5.3.1). We train it on ImageNet [136] and 8K+ images from our dataset, and three times on an annotated subset of 768 common failure cases (CFC) from our dataset. We evaluate and compare our model with YOLOv3 [108] and Faster R-CNN [109] on a test set of 2K+ images from our dataset. We use the regular MAP on all intended objects. Table 5.1 shows the
Table 5.1: The accuracy comparison between our object detector model versus the YOLOv3 and Faster R-CNN on our dataset to detect some known objects.

<table>
<thead>
<tr>
<th>Method</th>
<th>MAP</th>
<th>person</th>
<th>seat</th>
<th>plant</th>
<th>animal</th>
<th>car</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv3[108]</td>
<td>51.5</td>
<td>72.5</td>
<td>39.4</td>
<td>32.8</td>
<td>69.7</td>
<td>54.0</td>
</tr>
<tr>
<td>Faster R-CNN[109]</td>
<td>52.7</td>
<td>73.9</td>
<td>40.6</td>
<td>34.2</td>
<td>71.0</td>
<td>54.2</td>
</tr>
<tr>
<td>Ours</td>
<td>60.1</td>
<td>77.8</td>
<td>53.1</td>
<td>46.6</td>
<td>69.5</td>
<td>57.7</td>
</tr>
</tbody>
</table>

Table 5.2: The comparison between our pose estimator versus OpenPose and HRNet on our dataset to detect body parts.

<table>
<thead>
<tr>
<th>Method</th>
<th>MAP</th>
<th>hea</th>
<th>sho</th>
<th>elb</th>
<th>wri</th>
<th>hip</th>
<th>kne</th>
<th>ank</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenPose[110]</td>
<td>70.2</td>
<td>86.5</td>
<td>79.8</td>
<td>71.2</td>
<td>62.6</td>
<td>68.3</td>
<td>63.9</td>
<td>60.4</td>
</tr>
<tr>
<td>HRNet[111]</td>
<td>73.7</td>
<td>92.0</td>
<td>84.4</td>
<td>73.5</td>
<td>66.3</td>
<td>71.3</td>
<td>65.6</td>
<td>61.0</td>
</tr>
<tr>
<td>Ours</td>
<td>75.6</td>
<td>92.5</td>
<td>86.8</td>
<td>75.8</td>
<td>66.1</td>
<td>71.7</td>
<td>68.6</td>
<td>64.6</td>
</tr>
</tbody>
</table>

MAP and the average accuracy of some objects (person, seat, plant, animal, and car) for our trained model versus the YOLOv3 model. The “seat” average accuracy is the average for “seat, bench, and chair”, “plant” average accuracy is the average for “plant, tree, and grass”, and “animal” average accuracy is the average for “bird, cat, dog, cow, and sheep”.

5.6.2.2 Pose Estimator

We train our pose estimator model on MSCOCO[140], MPII[156], 4K+ images from our dataset, and three times on our 317 CFC. To evaluate the performance of our pose estimator model on a test set of 1K+ images from our dataset, we leverage MAP of all limbs. The comparison results of the MAP performance between OpenPose[110], HRNet[111], and our approach on a subset of 507 testing images from our dataset are shown in Table 5.2, where the left limb and the right limb are merged.

5.6.2.3 Scene Parser

To train our scene parser model, we use the ADE20K dataset [141], 4K+ images from our dataset, and 576 common failure cases annotated by LabelMe [157]. To evaluate scene parsing performance on a test set of 1K+ images from our
<table>
<thead>
<tr>
<th>Method</th>
<th>Pixel Accuracy (%)</th>
<th>Mean IoU (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSPNet[112]</td>
<td>74.9</td>
<td>40.8</td>
</tr>
<tr>
<td>DeepLab[113]</td>
<td>77.6</td>
<td>42.2</td>
</tr>
<tr>
<td>Ours</td>
<td>79.2</td>
<td>43.8</td>
</tr>
</tbody>
</table>

Table 5.3: The accuracy comparison between our scene parser versus PSPNet and DeepLab with 101-depth ResNet on our dataset.

Figure 5.7: The distributions of (a) the score obtained from the pose estimator, (b) the normalized area obtained from the pose estimator, and (c) the detection probability obtained from the object detector, and (d) the normalized area obtained from the object detector for our ground-truth images with a “person” as a common object by the pose estimator and the object detector.

dataset, pixel-wise accuracy (PixAcc) and mean of class-wise intersection over union (CIoU) are measured. The performance values of our scene parser model versus PSPNet [112] and DeepLab[113] with ResNet-101 are shown in Table 5.3 which indicates better PixAcc and CIoU is achieved on our dataset.

5.6.2.4 Hysteresis Fusion

The coverage of the photography ideas is improved by hysteresis fusion which allows the union of all images above HIGH thresholds across the detectors. We show how we configure these tunable thresholds, while we trade-off between the coverage and the accuracy across the object detectors. When we have more than one detector with common detectable objects, we fuse multiple features from the detectors to enhance common object detection. For example, “person” is a common object between object detector and pose estimation. We perform our pose estimator on our ground-truth images with a person or without any person from the dataset and calculate (a) the detection score (as mentioned in Eq. 5.3) and (b) the normalized area (i.e. the detected object area divided by the image area) of the dominant
Figure 5.8: For the ground-truth images without any “person”, the distributions of (a) the highest score (if any) obtained from the pose estimator, (b) the highest normalized area of the highest score object (if any) obtained from the pose estimator, and (c) the detection probability for the dominant object (if any) obtained from the object detector, and (d) the normalized area of the dominant object (if any) obtained from the object detector.

person (i.e. the person with the highest score) detected in each image as our pose estimator features. Also, we perform our object detector on those images and compute (c) the detection probability and (d) the normalized areas of the dominant person (i.e. the person with the highest probability) detected in each image as our object detector features.

The distributions of the features obtained from our pose estimator and our object detector for “person” as a common object for the pose estimator and the object detector has been shown in Figure 5.7. In some images, no person is detected by the pose estimator and the object detector, because the pose estimator or the object detector have a detection error or there is no person in the image. We consider such detection as non-person object detection. Figure 5.8 shows the distributions of those features obtained from our pose estimator and our object detector when there is no person in our ground-truth images, but they detect a person. We have removed the frequency of the first component, i.e., score or area = 0, from all of the curves in Figure 5.8, because the probability of zero score/area is very high and we want to bold the probabilities of the other score/area values.

Figure 5.7a shows pose estimator’s score does not have enough sensitivity to detect a person because the distribution is similar to a uniform probability mass function (PMF). Similarly, Figure 5.7d shows object detector’s normalized area does not have enough sensitivity to detect a person, because the distribution is pretty uniform. But, the object detector’s probability in Figure 5.7c and the pose estimator’s normalized area in Figure 5.8b are not similar to a uniform distribution, and we
can infer the cut-off thresholds from them. First, we derive the 2D probability
density function (PDF) of these mutual features including the normalized area by
the pose estimator and the detection probability by the object detector. Second,
we determine the 2D MAP surface w.r.t these two parameters as a heat map.
Finally, we search on the heat map to find the optimal point for these two mutual
features. As shown in Figure 5.9, it can be inferred from the 3D histogram of these
two features that the optimal HIGH cut-off thresholds are the object detector’s
probability 40% and pose estimator’s normalized area 10%. Similarly, the LOW
thresholds are object detector’s probability 28% and pose estimator’s normalized
area 4.5% that leads to the dataset coverage 84.7% and the detection accuracy
92.02% which is higher than any other detector accuracy solely.
Our CaDe  & 96.13 & 94.52 & 89.80 & 61.73 & 80.42 & 84.64 & 75.02 & 78.57 & 92.93 \\
16-feat Baseline & 66.58 & 80.24 & 68.35 & N/A & N/A & N/A & N/A & 59.42 & 55.28 \\

| Table 5.4: The accuracy results of our category detector (CaDe) for ground-truth images compared to a 16-feature-based baseline. |

5.6.2.5  Portrait Category Detection

As mentioned in Section 5.3.2, we start with top-down hierarchical clustering to specify the genre of the input image, and then we do multi-class categorization for portrait images. We train our model having 40 suggested features on a set of 20K+ annotated portraits from our dataset, and we test the model on another set of 5K+ annotated portraits. The mean average accuracy of the model is listed in Table 5.4 categorized by various styles.

Also, we just consider the first 16 features for object detectors including general max and number of detected people in the image as mentioned in Section 5.3.2 and train a model using the same ground truth as before. The current model is our baseline model because it can be used for any other object detector, as the features can be defined in other object detector domains as well. To compare rationally with this baseline, we test the same set of images from our ground truth. The second line in Table 5.4 listed the baseline results. Because we remove limb features, the baseline cannot detect sub-genres such as hand-only, leg-only, no-face, and side view.

5.6.2.6  Artistic Pose Clustering

Regarding artistic pose clustering, we conduct an experiment to cluster similar professional poses using our features explained in Section 5.3.3. We do the clustering with a various number of cluster heads, and we find the optimal number of cluster heads for our dataset using the elbow method [146]. That being said, we use the elbow method and do the clustering 40 times with the different number of clusters ranged from 1 to 40. This method calculates the sum of squared errors (the distance of each point to the center of its cluster) and it is expected to see an elbow pattern in the plot of this error when the number of clusters is increasing. The result of this method on our dataset is depicted in Figure 5.10, which indicates that the best choice for the number of clusters in this dataset is between 13 and 17. Since we
<table>
<thead>
<tr>
<th>Extractor</th>
<th>Effect</th>
<th>Baseline</th>
<th>% over baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOD</td>
<td>object-aware</td>
<td>max(detectors)</td>
<td>17.30</td>
</tr>
<tr>
<td>CaDe</td>
<td>category-aware</td>
<td>16-feat version</td>
<td>32.54</td>
</tr>
<tr>
<td>ArPose</td>
<td>pose-aware</td>
<td>DEC</td>
<td>25.7</td>
</tr>
</tbody>
</table>

Table 5.5: The summary of the methods used in CAPTAIN with respect to extraction algorithm, effect on retrieval, and improvement over baseline.

![Elbow Method](image)

Figure 5.10: The result of the elbow method on the dataset. We could spot the elbow around 13-17 clusters. We are also showing the first derivative of the distortion, to show where it is going to flatten out.

integrate many features with different extraction algorithms, Table 5.5 summarizes these features in aspects like extraction method, effects, the improvement over baseline.

### 5.6.3 User Study for Composition

There exists no directly similar or comparable system in the literature to compare with our proposed framework. The studies [102, 105, 158–161] in related work have different goals because they should get a known landmark with its meta-data to extract photos with similar geo-location, weather, and time, and process them to find the best view or camera parameters. But we retrieve exemplars from our 500K+ portrait and landscape dataset based on aesthetics-aware primitives of the given photo including IOD semantics, VGG generic, CaDe, pose, and gender.
To fairly evaluate the functionality and performance of our method, and measure how much the recommended photos are relevant to the query and helpful to the photographer, we conduct a quantitative user study to compare our method with other reasonable approaches. The first method directly finds good composition from scenes (called Chang’s method [1]). The second method is a retrieval method based on the color, shape, and texture features (called Mitro’s method [134]). The third and fourth ones are from a state-of-the-art semantic and scene retrieval method [162] with better convolutional networks (CNN) including the VGG-19 model [114] and ResNet-152 [115]. To create the last two baselines, all generic descriptors of the last pooling layer pre-trained on ImageNet [136] are evaluated for our dataset images as well as the features of the non-CNN-based method [134], and it is used as feature matrix $M_{\text{feat}}$ in Eq. 5.20. The similarity scores and normalization are calculated following the composition step (section 5.4), and user preferences are specified.
uniform across all competitors. The qualitative results of the composition step for some queries with user-specified preferences (USP) are illustrated in Figure 5.11. Each row includes query image, results from method [1], our method for equal weights, and ours for USP-aware retrieval.

We select a variety of image queries (Figure 5.12) based on background scene and semantics, single versus group, full-body, upper-body, facial, standing versus sitting, and male versus female. We do not use USP-aware queries as shown in Figure 5.11, and we focus on the diversity of image queries. Also, the same question throughout the study is asked, and therefore, we do not convey any side information. Using a PHP-based website with usage guidance, the outputs of the methods are randomly shown in each row to be chosen by 103 participants.

The expected value of the accepted recommended photos by the participants with respect to the total number of recommendations including the baselines is 65.74%. More accurately, the histogram of the acceptability rate for the queries of the user study is shown in Figure 5.13. The x-axis shows the acceptability rate ranged from 0 to 1 with 0.1-width bins, i.e., what percentage of the participants has accepted our recommended photos for those queries. The y-axis shows the frequency of our accepted recommendations by the total number of the examined corresponding queries (i.e. probabilities) which fall into each bin. The histogram has indicated that 16.07% of our recommended photos were accepted by over 80% of the participants, 67.86% of them with over 60%. Consequently, the majority of our recommended photos are accepted with a mean of 65.74%.
Figure 5.13: The histogram of the acceptability rate based on the recommended photos versus the total number of recommendations (recommendation probability) compares ours with other methods.

5.6.4 Runtime Analysis

We continuously collect images with various styles and categories for our dataset. We mostly used an NVIDIA Tesla K40 GPU for training and testing purposes, which took a couple of days for the intensive computations on the dataset. So, offline processing (indexing) of the 500px dataset is not included in the runtime results. The decomposition step is the bottleneck when we analyze the end-to-end runtime performance of our method. The decomposition step (including IOD, CaDe, and ArPose) is performed as three parallel processes on query images to measure the average duration of the decomposition step. We then perform PAIR in the composition step to get a ranked list of retrieved results for each image. Also, we randomly select the preferred style set and perform the matching algorithm for each image as a single available shot. We sum up the average time for these sequential steps to get the average runtime. The runtime is roughly proportional to the decomposition step because the internal IOD process that uses deep-learned models for detection is time-consuming. Among the detectors in IOD, the pose estimator
takes longer than the others and is pretty independent of the number of people in the image, with an average duration of around 94.7 milliseconds. Consequently, using a parallel structure shown in Figure 5.2, the end-to-end runtime is around 103.5 ms.
Chapter 6  
Conclusions and Future Work

6.1 Conclusions

We studied the realization of having a comprehensive framework to help amateur photographers to take better shots. First, we started with vanishing point detection to create a perspective-aware exemplar retrieval to assist people. Then, we exploited another photography principle called the triangle technique to retrieve triangular exemplars, especially for portraits. Then, we came up with an intelligent portrait composition assistance to extract aesthetics-related features of a portrait image and similarly give a highly-rated photograph. Finally, we extended our work to comprehensive composition assistance. The proposed framework detects aesthetics-related ingredients in query shots and retrieves more related photos based on the user-specified preferences and aesthetics-related information. We have experimented with the proposed approach using a large dataset for portrait and landscape photography ideas that we have collected. This study leverages the integration of deep-learning-based detectors, hysteresis fusion, portrait categorization, and artistic pose clustering, which makes the whole process automatic. As the number of photography ideas increases, retrieving the exemplars from the dataset becomes more challenging. Furthermore, the retrieval system finds similar images and searches for images with similar semantic constellations with better composition through decomposition and composition steps. After providing feedback for the photographer, the final pose is matched with the retrieved feedback and helps the photographer take a better shot. The performance of our framework has been
evaluated by a set of experiments, including comparisons with some competitors and a user study. However, we believe that our work opens up new research avenues to improve the proposed approach.

6.2 Future Directions

We explained several directions to accommodate the need for taking astonishing photos. The first and second approaches were based on commonly known photography rules, and the third and its extended version generalized the system having any available photography idea. However, there are other directions to extend the work, as we discuss in the following sub-sections.

6.2.1 Genre Extendibility

The general idea behind this work can be extended to other photography genres such as candid, fashion, close-up, and architectural photography using other appropriate detectors. The criteria for one genre are generally different from those for another. For instance, the pose is crucial in portrait photography, while leading lines and vanishing points can be crucial in architectural photography.

6.2.2 Enhancement in User Interaction

The individual should quantify the user-specified preferences (USP), but it may be difficult for them to accurately adjust the importance of the detectors for their personal preference. They may want to make a hierarchical selection, as some results are eliminated in each branch when going down the user-specified decision tree. Qualitatively they check the results and develop a better decision, but it may be time-consuming for them. One can optimize the decision weights for a specific user after learning his/her behavior, and then they can request the ordering (not the weight values). We believe that there is still room to improve the interactions with the individual.
6.2.3 Clustering of Photography Ideas

Some future directions include working on an unsupervised learning approach that can cluster all the images based on various photography ideas. Recognizing the ideas is not easy for amateurs, and one shot can have multiple ideas. After clustering, we might detect new ideas. Therefore, it would be interesting to explore a metric to estimate the potential novelty of the current shot based on computing similarity to other shots.

6.2.4 Innovative Shot

Another promising idea is to design a system where the camera automatically detects an innovative situation and takes a shot. Conventional methods in machine learning use the history of the field to help amateurs take professional photos, and of course, these approaches can not go beyond it. However, after recognizing new photography ideas, the system can take it as a compact space, not a finite discrete space, and it attempts to find a solution in this compact space. Fortunately, the complexity of the problem can change from integer programming to linear programming. Still, the way we define these compact spaces is hard based on the complexity of finding new photography ideas.
Bibliography


IEEE, pp. 3001–3008.


Transactions on Pattern Analysis and Machine Intelligence, 35(12), pp. 2821–2840.


Graphics and Interactive Techniques in Asia, Association for Computing Machinery, New York, NY, USA, pp. 1–10.


pp. 217–220.


Vita
Farshid Farhat

Contact: farshid83 AT gmail DOT com
Other Links: LinkedIn, GSite, GitHub, GitLab.

Education:
- PhD candidate, School of Electrical Engineering and Computer Science
  The Pennsylvania State University
  Co-Advisers: Professor James Z. Wang and Professor Jesse Barlow
  Thesis Title: Comprehensive Photographic Composition Assistance Through Meaningful Exemplars.
- MS, Computer Science and Engineering
  The Pennsylvania State University, Microsystems Design Lab (MDL)
  Thesis Title: Stochastic modeling and optimization of stragglers in MapReduce framework.
- PhD, Electrical Engineering
  The Sharif University of Technology, Electronics Research Center
  Adviser: Professor Shahrokh Ghaemmaghami
  Thesis Title: Image Steganalysis of Low Rate Embedding in Spatial Domain.
- MS, Electrical Engineering
  The Sharif University of Technology, Intelligent Systems Research Lab
  Adviser: Professor Mohammad Reza Aref
  Thesis Title: Analysis of Routing Misbehavior in Adhoc Networks.
- BS, Electrical Engineering
  The Sharif University of Technology
  Adviser: Professor Javad Salehi
  Thesis Title: Analysis of Dynamic Bandwidth Allocation Algorithms in Fiber to the Home.

Experience:
- Roboticist at Dexterity Inc.
- Chief Executive Officer at Simplr Technologies.
- Visiting Researcher at Chinese Academy of Sciences.
- Visiting Researcher at Microsoft.
- Software Engineer Intern at Western Digital.
- Research and Teaching Assistant at The Pennsylvania State University.