This work titled "Learning Based Image and Video Editing" by LEVENT KARACAN has been approved as a thesis for the Degree of DOCTOR OF PHILOSOPHY IN COMPUTER ENGINEERING by the below mentioned Examining Committee Members.

Prof. Dr. M. Önder EFE
Head

Assoc. Prof. Dr. M. Erkut ERDEM
Supervisor

Prof. Dr. Pınar DUYGULU ŞAHİN
Member

Assoc. Prof. Dr. A. Oğuz AKYÜZ
Member

Assoc. Prof. Dr. Tolga ÇUKUR
Member

This thesis has been approved as a thesis for the Degree of Doctor of Philosophy in Computer Engineering by Board of Directors of the Institute of Graduate School of Science and Engineering on ....../......

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU
Director of the Institute of Graduate School of Science and Engineering
To my lovely wife and soon-to-be-born daughter...
ETHICS

In this thesis study, prepared in accordance with the spelling rules of Institute of Graduate School of Science and Engineering of Hacettepe University,

I declare that

• all the information and documents have been obtained in the base of the academic rules.
• all audio-visual and written information and results have been presented according to the rules of scientific ethics
• in case of using others works, related studies have been cited in accordance with the scientific standards
• all cited studies have been fully referenced
• I did not do any distortion in the data set
• and any part of this thesis has not been presented as another thesis study at this or any other university.

05/12/2019

LEVENT KARACAN
YAYINLAMA FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanma açma iznini Hacettepe üniversitesine verdiği bildirim. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkîyet haklarımız bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarında (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğini ve tezimin tek yetkili sahibi olduğunu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alarak kullanıldığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeye taahhüt ederim.


☐ Enstitü yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihinden itibaren 2 yıl ertelenmiştir.

☐ Enstitü yönetim kurulu gerektiğinde tezim erişime açılması mezuniyet tarihinden itibaren.... ay ertelenmiştir.

☐ Tezim ile ilgili gizlilik kararı verilmiştir.

05/12/2019

LEVENT KARACAN
ABSTRACT

LEARNING BASED IMAGE AND VIDEO EDITING

Levent KARACAN

Doctor of Philosophy, Computer Engineering Department
Supervisor: Assoc. Prof. Dr. M. Erkut ERDEM
December 2019, 151 pages

Image and video editing encompasses the wide range of image operations to give desired visual effects to a given image or video either for improving various visual properties such as color, contrast, luminance or better emphasizing some aspects of scenes such as objects, background, activity, attribute, emotion etc. Popular graphical tools (e.g. Adobe Photoshop, GIMP) that provide rich image operations can be utilized to achieve the desired visual effects, however users need to be familiar with image processing methods and have skills to overcome challenging low-level operations on images and videos. Therefore, easy and efficient image and video editing methods are needed for casual users to manipulate visual contents with high-level interactions such as natural languages. On the other hand, it is expected that the processes will be imperceptibly flawless on the image, in other words, photorealism should not be degraded. Recently, data-driven or learning based new works which try to meet those expectations have been proposed for various image and video editing problems. In this thesis, we propose learning-based methods for a number of image and video editing problems which are alpha matting, visual attribute manipulation and language-based video manipulation following recent trends and developments. Our methods produce competitive or better results against state-of-the-art methods on benchmark datasets quantitatively and qualitatively while providing simple high-level interactions such as natural language and visual attributes. Besides, our visual attribute manipulation method is the first
high-level photo editing approach to enable continuous control on transient attributes of natural landscapes in the literature.

For alpha matting, we present a new sampling-based alpha matting approach for the accurate estimation of foreground and background layers of an image. Previous sampling-based methods typically rely on certain heuristics in collecting representative samples from known regions, and thus their performance deteriorates if the underlying assumptions are not satisfied. To alleviate this, we take an entirely new approach and formulate sampling as a sparse subset selection problem where we propose to pick a small set of candidate samples that best explains the unknown pixels. Moreover, we describe a new dissimilarity measure for comparing two samples which is based on KL-divergence between the distributions of features extracted in the vicinity of the samples. The proposed framework is general and could be easily extended to video matting by additionally taking temporal information into account in the sampling process. Evaluation on standard benchmark datasets for image and video matting demonstrates that our approach provides more competitive results compared to the state-of-the-art methods.

For visual attribute manipulation, we explore building a two-stage framework for enabling users to directly manipulate high-level attributes of a natural scene. The key to our approach is a deep generative network which can hallucinate images of a scene as if they were taken at a different season (e.g. during winter), weather condition (e.g. in a cloudy day) or time of the day (e.g. at sunset). Once the scene is hallucinated with the given attributes, the corresponding look is then transferred to the input image while preserving the semantic details intact, giving a photo-realistic manipulation result. As the proposed framework hallucinates what the scene will look like, it does not require any reference style image as commonly utilized in most of the appearance or style transfer approaches. Moreover, it allows to simultaneously manipulate a given scene according to a diverse set of transient attributes within a single model, eliminating the need of training multiple networks per each translation task. Our comprehensive set of qualitative and quantitative results demonstrate the effectiveness of our approach against the competing methods.

In our last work, we introduce a new task of manipulating person videos with natural language, which aims to perform local and semantic edits on a video clip of an individual to automatically change their outfit based on a description of target look. To this end, we first collect a new video dataset containing full-body images of different persons wearing different types of clothes and their textual descriptions. The nature of our problem allows for
better utilization of multi-view information and we exploit this property and design a new language-guided video editing model. Our architecture is composed of two subnetworks trained simultaneously: a network for constructing a concise representation of the person from multiple observations (representation network), and another network that benefits from the extracted internal representation for performing the manipulation according to the target description (translation network). Our qualitative and quantitative evaluations demonstrate that our proposed approach significantly outperforms existing frame-wise methods, producing temporally coherent and semantically more meaningful results.

**Keywords:** Image editing, Video editing, Alpha Matting, Attribute Manipulation, Language-based editing
ÖZET

ÖĞRENME TEMELLİ GÖRÜNTÜ VE VIDEO DÜZENLEME

Levent KARACAN

Doktora, Bilgisayar Mühendisliği
Danışman: Doç. Dr. M. Erkut ERDEM
Aralık 2019, 151 sayfa

Alfa matlama için, bir görününün ön alanları ve artalan katmanlarının yanlışız tahmini için yeni bir örneklemе tabanlı alfa matlama yaklaşımı sunuyoruz. Önceki örneklemе tabanlı yöntemler bilinen bölgeden temsil örnekler toplamak için genellikle belli sezgisellere dayanmaktadır, ve bu nedenle eğer dayanılamaz varsayımlar gerçeklemezse başarım kötü etkilenmektedir. Bunun üstesinden gelebilmek için, tümüyle yeni bir yaklaşım benimsiyoruz ve aday kümesinden bilinmeyen pikselleri en iyi ifade eden küçük aday örnekler koşmesini seçmek için önerdikimiz seyrek altkısma katman arasındaki KL-diversity'a dayalı iki örneklemeye imkan veren iki aşamalı bir model sunuyoruz. Ayrıca, örneklerin etrafında çıkarılan özellik dağılımları arasındaki KL-diversity'a dayalı iki yöntem genel bir yaklaşım ve örneklemeye aşamasında uzamsal bilgi de dikkate alınarak video matlamaya kolayca genişletilebilmektedir. Standart karşılaştırmalı yaklaşımlar ve aday kısım seçmek için yeni bir Alfa matlama icin, bir görününün yüksek düzeyli niteliklerini doğrudan düzenlemeye imkan veren iki aşamalı bir yöntem çatısı arastırıyoruz. Ayrıca, önerilen çatısı araştırıcı görüntüler sani bir başka semada(örneğin kış sırasında), hava durumunda(örneğin bulutlu biründe) veya günün zamanında(örneğin gün batımında) çekilmiş gibi bir sahnenin görüntülerini snarlayabilen denir üreticinin ağdır. Sahne verilen niteliğe göre snarlanıktan sonra, ilgili görünüş sonrasında anlamsal ayrıntıları koruyarak ve foto-gerçekçik düzenlemeye sonucu vererek girdi görünüşüne taşınır. Önerilen yöntem çatısı sahnenin farklı nitelikleri görünümünü snarlayabilmesi için, birçok görünüş ve stil taşınma yaklaşımlarında kullanıldığı gibi referans bir görünüş gerektirmez. Üstelik, yöntemimiz her çevirim görevi için aynı bir ağ eğitme ihtiyacını ortadan kaldırarak, tek bir model içinde farklı geçici niteliklere göre eş zamanlı olarak verilen sahneyi düzenlemeye olanak sağlar. Kapsamlı nitel ve nicel sonuçlarımız, rakip yöntemlere karşı yaklaşımımızın etkililiğini göstermektedir.

Son çalışmadımda, hedef görünüş tanımına dayalı otomatik kıyalet değişim yapmak için birisinin video kliyi üzerinde otomatik olarak yerel ve anlamsal düzenleme yapmayı amaçlayan yeni bir doğal dille insan videoları düzenleme görevi tanıttımyoruz. Bunun için, özellikle farklı türde kıyaflar giren farklı insanların tüm vücut görüntülerini ve bunların metinlerin tanımlamalarını içeren yeni bir video veri kümesi topluyoruz.Problemin doğası çoklu-özellikli(multi-view) bilgiden daha iyi yararlanmak için olacak sağlanmıştır, bu özelliği kullanımyoruz ve yeni bir doğal dille yönlendirilen video düzenleme modeli
tasarlıyoruz. Mimarımız eş zamanlı eğitten iki alt ağdan oluşmaktadır: çoklu gözlemden insanın özet temsilini oluşturmak için bir ağ(temsil ağı) ve hedef tanıma göre düzenleme gerçekleştirerek için çıkarılan üç temsiden başka bir ağ(çevirim ağ). Nitel ve nicel değerlendirmelerimiz önerdigimiz yaklaşımın daha anlamsal ve zamansal olarak uyumlu sonuçlar üretmek tek görüntü tabanlı yöntemleri önemli derecede geçtiğini göstermektedir.

**Keywords:** Görüntü düzenleme, Video düzenleme, Alfa Matlama, Nitelik Düzenleme, Dil tabanlı düzenleme
ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my supervisors Assoc. Prof. Dr. Erkut ERDEM and Assoc. Prof. Dr. Aykut ERDEM for their great guidance. They continuously supported this thesis with their experiences and knowledge and encouraged me to do my best all the time. I am aware that we together set the bar higher for me.

Besides, I would like to thank to my thesis committee members, Prof. Dr. M. Önder EFE, Prof. Dr. Pınar Duygulu ŞAHİN, Assoc. Prof. Dr. A. Oğuz AKYÜZ and Assoc. Prof. Dr. Tolga ÇUKUR for reviewing this thesis and giving insightful comments.

I would like to thank my friends, Emre AYDOĞAN, Cemil ZALLUHOĞLU, Selim YILMAZ, Handan GÜRSOY DEMİR and all research assistants for good days in Hacettepe University. I would also like to thank, Çağdaş BAŞ, Aysun KOÇAK and other members of Computer Vision Lab (HUCVL) for their collaboration and fun atmosphere. I would also like to thank Mert KILIÇKAYA for his hospitality and friendship at University of Amsterdam.

Furthermore, i would like to thank Prof. Dr. Zeynep AKATA for accepting me in Amsterdam Machine Learning Lab (AMLab) for short term research visiting.

I would like to express my deepest gratitude to my parents, Gülizar (Güllü) and Behsat. It was not possible to do this thesis without their support and encouragements.

Lastly, i would especially like to thank my devoted and beautiful wife, Emine, very much for her love and support. Nowadays, we are expecting our sweet daughter to come into the world in a few weeks. I don’t know how to express my gratitude and love to my wife, but i know how wonderful mother she will be.

I would also like to thank the Scientific and Technological Research Council of Turkey (TÜBİTAK) for supporting this thesis with 2214-A International Research Fellowship Program, 1059B141700309 and the research grants 112E146 and 217E029, and Turkish Academy of Sciences (TÜBA) with the Young Scientists Award (TÜBA-GEBIP) to Erkut Erdem.
GENİŞLETİLMİŞ ÖZET


Alfa matlama verilen görüntüde kullanıcı tarafından kabaca verilen artalan ve önalan bilgilerini kullanarak hassas bir şekilde önalanın aralanından ayrılanması problemi olarak tanımlanmaktadır. Alfa matlama ile hassas biçimde ayrılan önalan farklı aralanlara yerleştirilerek yeni görüntü ve video üretmekte kullanılmaktadır. Bu nedenle popüler grafik


Son çalışmadan da, hedef görüntü tanımlama dayalı otomatik kıyafet değişimi yapmak için bir kişinin video klibi üzerinde otomatik olarak yerel ve anlamsal düzenleme yapmayı
amaçlayan yeni bir doğal dille insan videoları düzenlemeye görevi tanıttırız. Doğal dil tanımına bağlı görüntü üretmek için önerilen ilk koşullu Çekişmeli Üretici Ağı modeli [38] dilsel tanımlamaları bulunan kuş ve çiçek verikümerleri üzerinde eğitikerek dilsel metinle verilen tanıma uygun yeni kuş ve çiçek görüntüü üretbilmektedir. Bu çalışmanın sonrasında aynı veri kümlerleri üzerinde çeşitli stratejilerle bu çalışmaya uyileştirir veya girdi görüntüüsü üzerinde düzenlemeye imkan sağlayan yeni çalışmalar [45, 46] önerilmiştir.

## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>i</td>
</tr>
<tr>
<td>ÖZET</td>
<td>iv</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>vii</td>
</tr>
<tr>
<td>GENİŞLETilmiş ÖZET</td>
<td>viii</td>
</tr>
<tr>
<td>CONTENTS</td>
<td>xii</td>
</tr>
<tr>
<td>TABLES</td>
<td>xvi</td>
</tr>
<tr>
<td>FIGURES</td>
<td>xxiii</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1. Dissertation Overview</td>
<td>4</td>
</tr>
<tr>
<td>2. ALPHA MATTING WITH KL-DIVERGENCE-BASED SPARSE SAMPLING</td>
<td>6</td>
</tr>
<tr>
<td>2.1. Problem Definition</td>
<td>7</td>
</tr>
<tr>
<td>2.2. Related Works</td>
<td>7</td>
</tr>
<tr>
<td>2.2.1. Previous work on sampling-based image matting</td>
<td>10</td>
</tr>
<tr>
<td>2.2.2. Previous work on video matting</td>
<td>12</td>
</tr>
<tr>
<td>2.2.3. Our contributions</td>
<td>13</td>
</tr>
<tr>
<td>2.3. Proposed Approach</td>
<td>14</td>
</tr>
<tr>
<td>2.3.1. Dissimilarity Between Two Samples</td>
<td>15</td>
</tr>
<tr>
<td>2.3.2. Sampling via Sparse Subset Selection</td>
<td>17</td>
</tr>
<tr>
<td>2.3.3. Selecting The Best ((F, B)) Pair</td>
<td>19</td>
</tr>
<tr>
<td>2.3.4. Pre- and Post-Processing</td>
<td>22</td>
</tr>
<tr>
<td>2.4. Extension To Video Matting</td>
<td>23</td>
</tr>
<tr>
<td>2.4.1. Motion Aware Temporal Sampling</td>
<td>23</td>
</tr>
<tr>
<td>2.4.2. Alpha Refinement</td>
<td>25</td>
</tr>
<tr>
<td>2.5. Experiments</td>
<td>26</td>
</tr>
<tr>
<td>2.5.1. Image Matting</td>
<td>26</td>
</tr>
<tr>
<td>2.5.2. Video Matting</td>
<td>31</td>
</tr>
<tr>
<td>2.5.3. Sparse Scribbles</td>
<td>34</td>
</tr>
</tbody>
</table>
2.5.4. Runtime Performance ...................................................... 35
2.6. Discussion ........................................................................... 36

3. MANIPULATING ATTRIBUTES OF NATURAL SCENES VIA HALLUCINATION ........................................................... 37
3.1. Introduction .......................................................................... 38
3.2. Related Work ......................................................................... 40
  3.2.1. Image Synthesis ................................................................. 40
  3.2.2. Image Editing ................................................................. 43
3.3. ALS18K Dataset ..................................................................... 45
  3.3.1. Dataset Collection ............................................................ 46
3.4. Attribute Manipulation Framework ........................................... 47
  3.4.1. Scene Generation .............................................................. 48
    Background ........................................................................... 49
    Proposed Architecture .......................................................... 50
    Improved Training of SGNs ......................................................... 53
    Relevant Negative Mining ...................................................... 53
    Layout-Invariant Perceptual Loss ........................................... 54
  3.4.2. Style Transfer ................................................................. 55
3.5. Results and Comparison ........................................................... 57
  3.5.1. Attribute and Layout Guided Scene Generation .................... 57
    Training Details ................................................................. 57
    Ablation Study ................................................................. 58
    Comparison with Image-to-Image Translation Models ............. 60
    Diversity of the Generated Images ........................................ 64
    Adding and Subtracting Scene Elements ................................ 65
  3.5.2. Attribute Transfer ............................................................. 66
    Effect of Post-Processing and Running Times ......................... 74
  3.5.3. Effect of Center-cropping ................................................... 75
  3.5.4. A Standalone GUI Prototype ............................................ 76
  3.5.5. Limitations ................................................................. 76

xiii
# TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Evaluation of matting methods on the benchmark dataset [50] with three trimaps according to SAD, MSE and Gradient error.</td>
<td>29</td>
</tr>
<tr>
<td>2.2</td>
<td>Evaluation of recent related matting methods on the benchmark dataset [50] with three trimaps according to SAD and MSE error.</td>
<td>30</td>
</tr>
<tr>
<td>2.3</td>
<td>Effect of spatial sampling (single frame) vs. temporal sampling (two frames) and optical flow information (OF) to alpha matte results on SSD scores.</td>
<td>32</td>
</tr>
<tr>
<td>2.4</td>
<td>Evaluation of matting methods on the benchmark dataset [51] with three trimaps according to SSDA, SSDdt and MESSDdt.</td>
<td>33</td>
</tr>
<tr>
<td>3.1</td>
<td>Architectural details of the generator and discriminator networks.</td>
<td>53</td>
</tr>
<tr>
<td>3.2</td>
<td>Ablation study. We compare visual quality with respect to Inception Score (IS) and Fréchet Inception distance (FID), attribute and semantic layout correctness in terms of average MSE of attribute predictions (Att. MSE) and segmentation accuracy (Seg. Acc.), respectively, via pre-trained models. Our SGN model trained with RNM and PL techniques consistently outperforms the others, including the setting with VGG-based perceptual loss.</td>
<td>60</td>
</tr>
<tr>
<td>3.3</td>
<td>Quantitative comparison of layout conditioned image synthesis approaches. Our model consistently outperforms others in both coarse and fine resolutions in terms of photorealism, as measured by IS and FID.</td>
<td>64</td>
</tr>
<tr>
<td>3.4</td>
<td>User study results for attribute manipulation. The preference rate denotes the percentage of comparisons in which users favor one method over the other.</td>
<td>71</td>
</tr>
<tr>
<td>3.5</td>
<td>Running time analysis showing the average run time (in seconds) of each component of the proposed model across various image resolutions.</td>
<td>75</td>
</tr>
</tbody>
</table>
Table 4.1. Quantitative comparison against the state of the art models. Our approach outperforms the existing methods by a large margin in terms of all evaluation measures. For IS and VS, higher is better. For FID and VFD, lower is better.

Table 4.2. Ablation study of the proposed semantic video manipulation framework.
FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Two views of the same scene with different visual attributes. For example, given scenes can be described with visual attributes of clear and cloudy respectively. Image taken from [41]</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Green screen technology to compose a new scene. (Image credit: Alice in Wonderland(Movie))</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>Non-parametric sampling-based matting approaches. Top row: An input image and the representative samples gathered by the Robust [52], Shared [13], Global [14], Comprehensive [16], and the proposed Sparse Sampling based matting methods. The unknown pixel, the foreground and background samples are shown in yellow, red and blue colors, respectively. Bottom row: Comparison of the estimated alpha mattes by the suggested approach and the state-of-the-art Comprehensive Sampling matting method [16].</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>Distance embedding visualizations using t-SNE method [53] clearly demonstrate that the proposed KL-divergence based dissimilarity measure provides a better discrimination between known foreground and background pixels than using the standard color distance.</td>
<td>17</td>
</tr>
<tr>
<td>2.3</td>
<td>Sampling via sparse subset selection. Candidate foreground and background samples are shown in red and blue, respectively.</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>Temporal sampling using sparse subset selection. Candidate foreground and background samples, which are respectively shown in red and blue color, come from the previous frame ( t - 1 ) and the current frame ( t ).</td>
<td>26</td>
</tr>
<tr>
<td>2.5</td>
<td>Effect of ( \gamma ) parameter on the performance. Plot shows average MSE values over all training images and all trimaps.</td>
<td>27</td>
</tr>
</tbody>
</table>
Figure 2.6. Effect of including local samples to the representative set obtained with the proposed sparse sampling scheme. Plot shows average MSE values over all test images for three types of trimaps. 28

Figure 2.7. Visual comparison of our approach with other sampling-based image matting methods. (a) Input image, (b) CWCT sampling [17], (c) Comprehensive sampling [16], (d) LNSP matting [54], (e) Sparse coded matting [18] and (f) Proposed approach. 28

Figure 2.8. Training sequences from the video matting benchmark dataset [51]. 32

Figure 2.9. Visual comparison between the proposed and the other best-performing video matting methods on the video benchmark dataset [51]. (a) Input video frames, (b) Closed Form [5], (c) Refine Edge Tool, (d) Learning Based [19] and (e) Proposed approach. 33

Figure 2.10. Alpha matting results with scribble based user input and the corresponding MSE scores estimated over the input images. 35

Figure 3.1. Given a natural image, our approach can hallucinate different versions of the same scene in a wide range of conditions, e.g. night, sunset, winter, spring, rain, fog or even a combination of those. First, we utilize a generator network to imagine the scene with respect to its semantic layout and the desired set of attributes. Then, we directly transfer the scene characteristics from the hallucinated output to the input image, without the need for a reference style image. 37

Figure 3.2. Sample semantic layout predictions for some images from Transient Attribute dataset Laffont et al. [41]. 47

Figure 3.3. Sample transient attribute predictions for some images from ADE20K dataset [55]. 48

Figure 3.4. Distribution of the most frequent object classes in our proposed ALS18K dataset (sorted by their number of occurrences). 49
Figure 3.5. Overview of the proposed attribute manipulation framework. Given an input image and its semantic layout, we first resize and center-crop the layout to $512 \times 512$ pixels and feed it to our scene generation network. After obtaining the scene synthesized according to the target transient attributes, we transfer the look of the hallucinated style back to the original input image.

Figure 3.6. Scene Generation Network (SGN). Our proposed CGAN architecture for generating synthetic outdoor scenes consistent with given layout and transient attributes.

Figure 3.7. Sample scene generation results. In these examples, the input layouts are from the test set, which are unseen during training and the transient attributes are fixed to the original transient attributes. Incorporating Relevant Negative Mining (RNM) and Perceptual Loss (PL) significantly improves the performance of the baseline SGN model in terms of both image quality as well as faithfulness of the end result to conditioned layouts and attributes. Moreover, the way we define our perceptual loss, as compared to commonly used VGG-based one, provides better and more photorealistic results.

Figure 3.8. Comparison of our SGN model against Pix2pix [39] and Pix2pixHD [40]. Each row shows the original image and the samples generated according to its corresponding semantic layout. Since our SGN model also takes into account a set of target transient attributes (only the top three most significant ones are shown here for the sake of simplicity), it can generate diverse and more realistic results than the other methods.

Figure 3.9. Comparison of the proposed SGN against Pix2pixHD. Given the input semantic layout, we show the reference image this semantic layout belongs to and the synthetic images generated by Pix2pixHD and our SGN model.

Figure 3.10. A screenshot of a sample question in our user study on scene synthesis.
Figure 3.11. (a) Age, (b) gender and (c) profession distribution of the participants of our user study on scene synthesis................................. 64

Figure 3.12. Modifying transient attributes in generating outdoor images under different weather and time conditions. Our model’s ability of varying with transient attributes contributes to the diversity and photorealism in its generation (more results can be found in the Supplementary Material). .......................................................... 65

Figure 3.13. The effect of varying the noise vector. For an example synthetically generated scene (left), we show close-up views from three different regions (middle) from samples obtained with only changing the random noise. Standard deviation of each pixel over 100 different realizations of the scene (right), which demonstrates that the random noise causes stochastic variations within the irregular or stochastic textural regions. ........................................................... 65

Figure 3.14. Gradually adding and removing elements to and from the generated images. We use a coarse spatial layout map (top left) to generate an image from scratch, and then keep adding new scene elements to the map to refine the synthesized images. Moreover, we also show how we can modify the look by conditioning on different transient attributes. 66

Figure 3.15. Sample attribute manipulation results. Given an outdoor scene and its semantic layout, our model produces realistic looking results for modifying various different transient attributes. Moreover, it can perform multimodal editing as well, in which we modify a combination of attributes................................................. 67

Figure 3.16. Comparison with [41]. In each row, for a given input image (first column), we respectively provide the results of [41] using their exemplar-based style transfer method (second column) and FPST [43] (third column) between retrieved images and input images, and the results of our method (last column) using FPST [43] between generated image by proposed SGN model and input image... 68
Figure 3.17. Comparison with base method of [41]. For given input image, (a) and (b) results of [41] using the exemplar-based style transfer method they proposed and DPST method [42] respectively between retrieved images and input images, (c) results of our method using DPST [42] between generated image by proposed SGN model and input image...

Figure 3.18. A screenshot of a sample question in our user study on attribute transfer.

Figure 3.19. (a) Age, (b) gender and (c) profession distribution of the participants of our user study on attribute transfer.

Figure 3.20. Season transfer to paintings. Source images: Wheat Field with Cypresse by Vincent van Gogh (1889), In the Auvergne by Jean-Francois Millet (1869) and Lourmarin by Paul-Camille Guigou (1868), respectively.

Figure 3.21. Our method can produce photorealistic manipulation results for different degrees of transient attributes. We give the results of both (a)FPST and (b)DPST photo style transfer methods.

Figure 3.22. Effect of post-processing. Top: a sample input image and “Autumn” attribute transfer results by our framework with DPST [42] and FPST [43], respectively. Bottom: the impact of various post-processing strategies on the final results. See Section 3.4.2. for the details.

Figure 3.23. Effect of center-cropping on manipulating a panoramic image.

Figure 3.24. Attribute manipulation results on panorama images.

Figure 3.25. A screenshot of the GUI prototype running our approach as a backend.

Figure 3.26. Example failure cases for our attribute manipulation framework, which are due to the visual quality of synthesized reference style image (top row) and failing of the photo style transfer method (bottom two rows).
Figure 3.27. Attribute manipulation results obtained with our framework by using DPST \([42]\) and FPST as the style transfer method. Given a natural scene image as input, we select a set of transient attributes and consequently obtain different images of the same scene reflecting the characteristics of these attributes. 80

Figure 3.28. Attribute manipulation results automatically generated by our framework by using DPST \([42]\) and FPST \([43]\) as the transferring mechanism. Given a natural scene image as input, we select a set of transient attributes and consequently obtain different images of the same scene reflecting the characteristics of these attributes. 81

Figure 4.1. Language guided manipulation of person videos. First row shows the input sequence together with the original description, and the second row presents the manipulation results for the given target text. 84

Figure 4.2. Schematic illustration of the proposed editing approach. (a) Source video sequence. (b) Representation network \(R\) accepts multiple observations to compute a multi-view representation of the person by aggregating individually extracted representations. Translation network \(T\) then uses this internal representation when manipulating input frames according to the target description. 86

Figure 4.3. Our video manipulation framework consists of two subnets: (a) representation network \(R\) which is responsible for extracting relevant person-specific features from multiple frames, and (b) translation network \(T\) which carries out the semantic editing process according to target text description by taking into account the extracted person representation. 89

Figure 4.4. Image-wise discriminator. 90

Figure 4.5. Qualitative comparison against existing semantic image manipulation approaches. 93
Figure 4.6. t-SNE visualizations of the person images obtained with (a) 20-dimensional embeddings learned by the $\beta$-VAE in our representation network $R$, (b) the first 10 dimensions that are passed to our translation network $T$, and (c) the last 10 dimensions (zoom in to see the details).

Figure 4.7. Latent traversals for learned factors of variation. (a) $\beta$-VAE trained alone. (b) $\beta$-VAE in our representation network trained jointly with the translation network. Traversal range is (-3, 3).
1. INTRODUCTION

Image and video editing can be defined as manipulating a given image or video with respect to different aspects of image properties such as color, texture, contrast, object, light, size, foreground, background, etc. Film and advertising industry and social media users demand new image and video editing methods which can be used from enhancing visual quality of photos to creating new virtual worlds using real and synthetic objects. Although popular graphical tools (Adobe Photoshop, GIMP) provide users rich image operations to manipulate given image or video sequence to give the desired visual effect in various form from simple contrast enhancement to complex object adding-removing, they require users to be familiar with basic image processing concepts and graphical tools which provide off-the-shelf filters and algorithms to perform common effects and manipulations. Therefore, it does not appeal to ordinary users. In this respect, there are two important points that need to be considered for image and video editing: 1. Providing easy editing and 2. Giving the desired effect accurately. With the spread of smartphones, everybody has been enabled to easily take photos or record videos. Therefore, in almost every moment thousands of visual content have been started to be shared on social media. In the meantime, image and video editing previously done by experts has started to become an easy-to-do job thanks to data-driven image and video editing approaches using increasing visual data.

Data driven image and video editing approaches provide a natural way for editing using simple and natural interactions. For example, suppose a user wants to manipulate a scene so that it becomes cloudy like in Figure 1.1. He either must read a Photoshop tutorial about how to create clouds or do an art work with the proper layers. On the other hand, he can just say “more cloudy” in a more natural way to perform the same task with high-level interaction. Such a high-level editing method requires a well-organized dataset and a machine learning scheme to associate high-level attributes with specific feature representations, e.g. blue attribute associated with certain RGB values.
As well as simplicity of ideal image or video editing tool, we also expect edited image or video to become as natural as possible in terms of its own visual characteristics and to contain as little artifact as possible. For example, in a fantastic movie like in Figure 1.2, video editing techniques create a new realistic scenes which are not actually real by changing scene background or foreground seamlessly. Similarly, when we want to give the effect of a well-known painter to our landscape photo, we not only expect a simple editing experience but also our photo to have same visual content with original photo and reflect painter’s stylization on it accurately. As a result, to accomplish such hard image editing operations, an artist can use graphical tools and handcraft but for casual users, it is quite challenging to give desired effect without artifacts and preserve the realism.

In this regard, there is a nuance between handcraft and imagination. In other words, we can imagine how a scene look like in different visual forms, but we all are not so talented that we can not easily edit a scene to look like our imagination. Data-driven or learning based image and video editing approaches are proposed to fill this gap so that a casual user can easily edit an image or video with simple low-level or natural high-level interactions.
In this dissertation, we study data-driven methods for alpha matting, visual attribute manipulation and language based video editing to provide easy and natural way for image and video editing while maintaining photorealism.

For alpha matting, we propose a more principled way to sampling representative background and foreground colors unlike existing works which require certain spatial assumptions on sampling region. Moreover, we define a new KL-divergence based contextual measure as an alternative to chromatic and spatial distance. Then we extend our sampling approach to video matting with additional temporal information.

For visual attribute manipulation, we first build a scene generation network(SGN) conditioned by transient attributes(e.g "cloudy", "sunset", "snow", etc.) and semantic scene layout on conditional Generative Adversarial Network(cGAN) framework to control both semantic layout and transient attributes so that SGN is able to generate semantically similar image with input image and hallucinate desired attributes. Then we explore a number of photo style transfer methods to transfer hallucinated visual attribute to original image. To this end, we curate a new dataset from two popular dataset. Moreover, we design an image editing tool to manipulate transient attributes of a given input landscape photo interactively.

Lastly, for manipulating person videos with natural language, we propose a new deep model composed of two coupled modules, a representation network and a translation network. We collected a new video dataset which comprises of full-body images of individuals and the textual descriptions of the clothes they wear.
We evaluate our methods both qualitatively and quantitatively to demonstrate how they beat or compete with state-of-the-art methods with several advantages. We also show first time in the literature that our visual attribute manipulation method allows for manipulating an input landscape scene as if it is taken under different conditions such as "snowy", "cloudy", "sunset", etc even enabling to play degree of each condition.

Below, we give an overview of this dissertation.

1.1. Dissertation Overview

In this dissertation, we propose three different approaches for alpha matting, visual attribute manipulation and manipulating person videos with natural language, respectively.

In Section 2, we present a new sampling-based alpha matting approach for the accurate estimation of foreground and background layers of an image. Previous sampling-based methods typically rely on certain heuristics in collecting representative samples from known regions, and thus their performance deteriorates if the underlying assumptions are not satisfied. To alleviate this, we take an entirely new approach and formulate sampling as a sparse subset selection problem where we propose to pick a small set of candidate samples that best explains the unknown pixels. Moreover, we describe a new dissimilarity measure for comparing two samples which is based on KL-divergence between the distributions of features extracted in the vicinity of the samples. The proposed framework is general and could be easily extended to video matting by additionally taking temporal information into account in the sampling process. Evaluation on standard benchmark datasets for image and video matting demonstrates that our approach provides more accurate results compared to the state-of-the-art methods.

In Section 3, we explore building a two-stage framework for enabling users to directly manipulate high-level attributes of a natural scene. The key to our approach is a deep generative network which can hallucinate images of a scene as if they were taken at a different season (e.g. during winter), weather condition (e.g. in a cloudy day) or time of the day (e.g. at sunset). Once the scene is hallucinated with the given attributes, the
corresponding look is then transferred to the input image while preserving the semantic
details intact, giving a photorealistic manipulation result. As the proposed framework
hallucinates what the scene will look like, it does not require any reference style image
as commonly utilized in most of the appearance or style transfer approaches. Moreover, it
allows to simultaneously manipulate a given scene according to a diverse set of transient
attributes within a single model, eliminating the need of training multiple networks per each
translation task. Our comprehensive set of qualitative and quantitative results demonstrate
the effectiveness of our approach against the competing methods.

In Section 4, to manipulate person videos with natural language, we first collect a new video
dataset containing full-body images of different persons wearing different types of clothes
and their textual descriptions. The nature of our problem allows for better utilization of
multi-view information and we exploit this property and design a new language-guided video
editing model. Our architecture is composed of two subnetworks trained simultaneously: a
network for constructing a concise representation of the person from multiple observations
(representation network), and another network that benefits from the extracted internal
representation for performing the manipulation according to the target description (transla-
tion network). Our qualitative and quantitative evaluations demonstrate that our proposed
approach significantly outperforms existing frame-wise methods, producing temporally
coherent and semantically more meaningful results.

Discussion In Section 5, we give a summary of contributions of this dissertation and discuss
future works for the image and video editing problems we address.
2. ALPHA MATTING WITH
KL-DIVERGENCE-BASED SPARSE SAMPLING

In this section, we first present a novel sampling-based image matting method for which we propose a new KL-Divergence-based dissimilarity measure for a sparse subset selection method [24]. Then, we extend our image matting method to video matting considering temporal smoothness¹. Previous sampling-based alpha matting approaches were based on a certain spatial assumptions before this work. Extensive quantitative and qualitative experiments show that proposed sampling-based alpha matting method produces better results than previous methods and proposed KL-Divergence-based measure provides better discrimination than mean chromatic distance between superpixels.

![Figure 2.1. Non-parametric sampling-based matting approaches. Top row: An input image and the representative samples gathered by the Robust [52], Shared [13], Global [14], Comprehensive [16], and the proposed Sparse Sampling based matting methods. The unknown pixel, the foreground and background samples are shown in yellow, red and blue colors, respectively. Bottom row: Comparison of the estimated alpha mattes by the suggested approach and the state-of-the-art Comprehensive Sampling matting method [16].](image)

¹Early version of this work was presented in International Conference on Computer Vision (ICCV 2015) [56]. Full version was published in IEEE Transactions on Image Processing (TIP 2017) [57].
2.1. Problem Definition

Accurate estimation of foreground and background layers of an image or video frames plays an important role for many image and video editing applications. In the computer vision literature, this problem is known as alpha matting, and mathematically, it refers to the problem of decomposing a given image or video frame $I$ into two layers, the foreground $F$ and the background $B$, defined in accordance with the following linear image composition equation.

\[ I = \alpha p F_p + (1 - \alpha p) B_p \]  

(1)

where $\alpha p$ represents the unknown alpha matte which defines the true opacity of each pixel $p$ and whose values lies in $[0, 1]$ with $\alpha p = 1$ denoting a foreground pixel and $\alpha p = 0$ indicating a background pixel. This is a highly ill-posed problem since for each pixel we have only three inputs but seven unknowns ($\alpha$ and the RGB values of $F_p$ and $B_p$). The general approach to resolve this issue for image matting is to consider a kind of prior knowledge about the foreground and background in form of user scribbles or a trimap to simplify the problem and use the spatial and photometric relations between these known pixels and the unknown ones. As for the video matting, estimating the alpha mattes of each frame is a more challenging task than single image matting since it requires both temporally coherent and spatially accurate maps.

2.2. Related Works

Image matting methods can be mainly categorized into two groups: propagation-based methods [3–10] and sampling-based methods [11–18]. The first group defines an affinity matrix representing the similarity between pixels and propagate the alpha values of known pixels to the unknown ones. These approaches mostly differ from each other in their propagation strategies or affinity definitions. Recent, the information-flow matting method [10] proposed a new propagation-based closed-form solution combining both local and non-local affinities.
It can alternatively be used at the alpha refinement step instead of Closed-form matting [5]. The latter group, on the other hand, collects color samples from known foreground and background regions to represent the corresponding color distributions and determine the alpha value of an unknown pixel according to its closeness to these distributions. Early examples of sampling-based matting methods [11, 12] fit parametric models to color distributions of foreground and background regions. Difficulties arise, however, when an image contains highly textured areas. Thus, virtually all recent sampling-based approaches [13–18] consider a non-parametric setting and employ a particular selection criteria to collect a subset of known $F$ and $B$ samples. Then, for each unknown pixel $p$, they search for the best $(F_p, B_p)$ pair within the representative samples, and once the best pair is found, the final alpha matte is computed as

$$\hat{\alpha}_p = \frac{(I_p - B_p) \cdot (F_p - B_p)}{\|F_p - B_p\|^2}. \quad (2)$$

The recent sampling-based approaches mentioned above also apply local smoothing as a post-processing step to further improve the quality of the estimated alpha matte.

For video matting, several researchers extend the existing image matting methods so that they can extract temporally coherent alpha mattes by using either user-generated or pre-defined trimaps along the video frames. Some of these approaches [58–60] automatically generate trimaps by using user interaction to segment foreground object and morphological dilation operation and then apply image matting methods to compute alpha matte. The methods [58, 61, 62] which do not directly use temporal information suffer from the temporal inconsistency, on the other hand, the ones [59, 63–68] utilize the temporal information present more temporally coherent alpha matte results. These methods differ from each other in terms of how they incorporate temporal information to compute alpha matte along the video sequences. For a more comprehensive up-to-date survey of image and video matting methods, we refer the reader to [69, 70].

Apart from the two main types of approaches, there are also some hybrid methods which consider a combination of propagation and sampling based formulations [54], or some supervised machine learning based methods which learn proper matting functions from
a training set of examples [19]. With coming of the era of deep learning, learning-based matting has recently attracted more attention. The first deep learning based matting method DCNN matting [20] that employs a CNN network to learn to combine results of existing alpha matting methods. Xu et al. [21] introduced a large-scale matting dataset to train a completely data-driven deep network takes in input image and trimap and produces alpha matte. AlphaGAN [22] proposed a Generative Adversarial Networks(GAN) approach for alpha matting to improve the network architecture of Xu et al. [21]. Very recently a learning-based sampling method [23] were proposed to divide learning-based alpha matting into easier color sampling tasks instead of directly estimating alpha matte.

The proposed matting approach belongs to the group of sampling-based methods which will be reviewed in the next subsection. Relying on a non-parametric formulation, these methods typically exploit different strategies to gather the representative foreground and background samples. Our observation is that all these strategies lack a strong theoretical basis, i.e. they require certain assumptions to hold to capture the true foreground and background colors, and moreover, they fail to adequately utilize the relationship between known and unknown regions. In contrast, our approach offers a more principled way to sampling by casting it as a sparse subset selection problem [24, 71], in which the resulting samples refers to a small subset of known foreground and background pixels that best explains the unknown pixels. In particular, sampling is formulated as a row-sparsity regularized trace minimization problem which solely depends on pairwise dissimilarities between known and unknown pixels, and for that, we propose a new KL-divergence based contextual measure as an efficient alternative to chromatic and spatial distances. Besides we extend this sampling approach to video matting by incorporating temporal information together with temporal matting Laplacian to provide temporal coherency. Finally, we demonstrate proposed sampling strategy is quite feasible for sparse user input as scribble.
2.2.1. Previous work on sampling-based image matting

Sampling-based models differ from each other in (i) how it collects the representative foreground and background samples, and (ii) how it selects the best \((F, B)\) pair for an unknown pixel. Mishima’s Blue-screen matting method [72] captures the image of a foreground object in front of a monochrome background. This setup allows efficient estimation of foreground and background distributions via clustering, and then alpha values of unknown pixels are estimated by considering their proximity to the extracted clusters. Another early work, the Knockout system [73], estimates true color values of the foreground and background layers of an unknown pixel by a weighted sum of nearby known pixels with the weights proportional to their spatial distances to the unknown pixel.

Robust matting [52], for an unknown pixel, collects samples from the known nearby foreground and background pixels. Among those samples, it then selects the pair that best fits the linear composting equation defined in Eq. (1). As the selection is carried by considering the color distortion, it provides more robust results than the Knockout system. However, since sampling depends only on the spatial closeness to the unknown pixels, as shown in Fig. 2.1., the true samples might be missing in the candidate set, decreasing the matting quality. In [74], it has been shown that using geodesic distances improves the results of this model to a certain extent. Shared matting [13] gathers representative samples from the trimap boundary, assuming that, for an unknown pixel, its true foreground and background color can be found at the closest known region boundaries. These pixels are defined as the boundary pixels that lie along the rays which are originated from the unknown pixel and that partition the image plane into disjoint parts of equal planar angles. Then, the best pair among those are used to estimate its alpha value w.r.t. an objective function that depends on spatial and photometric affinity. It falls short, however, when the rays do not reach the true samples. Weighted color and texture (WCT) sampling [15] and its comprehensive version (CWCT) extend Shared matting by combining the local sampling strategy in [13] with a global one that depends on a clustering-based probabilistic model. Moreover, it uses a texture compatibility measure in addition to the color distortion measure to prevent selecting overlapping samples.
Global sampling [14] also collects samples from the trimap boundaries but to avoid the problem of missing true samples, instead of emanating rays from unknown pixels, as in [13], it considers all known boundary samples as a global candidate set. To handle the large number of samples, it employs a simple objective function and an efficient random search algorithm in finding the best sample pair. However, as shown in Fig. 2.1., the true colors might still be missed in the resulting sample set if they do not lie along the trimap boundaries.

Comprehensive sampling matting [16] follows a global strategy and divides the known and unknown regions into a number of segments so that the segment over which the samples are gathered is decided according to the distance of a given unknown pixel to the extracted foreground and background segments. Sample colors are constructed as the means of the color clusters that are obtained via a two-level hierarchical clustering modeled by a parametric Gaussian mixture model. This approach gives better results than the previous non-parametric sampling based approaches. However, there is still a possibility of missing true samples since the sampling strategy depends on spatial closeness. As demonstrated in Fig. 2.1., the true color samples might be very far away from the unknown pixel.

Sparse coded matting [18] formulates image matting as a sparse coding problem. It computes alpha values from a bunch of sample pairs within a sparse coding framework instead of finding only the best but single pair of foreground and background \((F, B)\) pair. These samples forming the dictionary atoms are collected from the mean color of the superpixels that lie along the boundaries of the trimaps. Thus, it might also suffer from the missing true samples problem. This problem is solved in its extended version[75] by including samples from whole image region. To prevent overlapping color distributions of foreground and background, it adaptively controls the dictionary size according to a confidence value that depends on probabilistic segmentation. A similar sparse coding approach is used in [76] while selecting samples via a two-level hierarchical k-means clustering process.
2.2.2. Previous work on video matting

Widely used blue screen matting [77] provides effective video mattes as extracting the foreground objects from a solid color background is easy, but it requires special studio environment. For natural backgrounds, classical video matting approaches [58, 59, 61, 62, 64, 66] first segment the foreground object from the background and construct a trimap, which will accordingly be propagated along the video frames and used as inputs to single image matting methods. The existing models in general differ from each other in terms of either their segmentation and trimap construction strategies or the latter considered matting schemes.

In particular, Chuang et al. [61] propose a video matting method which builds upon Bayesian matting [11] in extracting the foreground layer. Li et al. [58] generate a binary mask via a Graph-Cut based segmentation algorithm, which will be used as input for Coherent matting [78]. Wang et al. [59] employ a Mean-Shift segmentation approach to segment the foreground objects in video sequences. Video SnapCut [60] proposes a new interactive video object extraction system using localized classifiers for local image features such as color, edge and learned shape prior. To impose temporal coherency, it considers the alpha matte computed from previous frame as a prior for the current frame. Tang et al. [64, 66] compute a probability map or an opacity map and again construct a Graph-Cut formulation to segment the video frames into foreground and background layers before applying a 3D Closed-form matting extended from [5]. Bai et al.[62] ask the user to refine the automatically extracted trimaps on some keyframes and then by using optical flow information between frames they propagate these trimaps to all video frames which are later used as inputs to Robust matting [52].

Video matting models which have been recently proposed are particularly focused on the matting part of the pipeline and interested in extracting temporally more coherent alpha mattes. An important direction here is to extend the matting Laplacian [5] by temporal information. For example, [64–68, 79] all employ a matting Laplacian extended to 3D by additionally considering the temporal domain. This modification provides extra local...
smoothness over extracted alpha mattes. Specifically, Choi et al. [65] use Non-local matting [6] approach to define a 3D nonlocal matting Laplacian on a 3D nonlocal neighborhood between video frames to propagate alpha matte values along the video sequences. Li et al. [67] incorporate motion information to KNN Laplacian [7] by using two-frame affinity matrix and propose a closed-form solution. Shahrian et al. [68] propose an improvement over the Comprehensive sampling scheme [16] in which the sampling phase is expanded by considering previous frame samples and by using local texture features to provide temporal and spatial consistency. Finally, they apply a temporal refinement via 3D matting Laplacian and the alpha matte priors computed from the previous frames. Zou et al. consider the non-local principles in [65, 67] and formulate a sparse dictionary learning problem to represent the known foreground and background colors provided from user input. Similarly, a refinement procedure is applied as a final step by employing a two-frame matting Laplacian.

2.2.3. Our contributions

As described, all the existing sampling-based image matting methods rely upon different assumptions regarding the selection policy of background and foreground samples. The justification of these assumptions are mostly valid. But still, they are heuristic methods and they all lack a theoretical ground to explain the relationship between known and unknown pixels in all possible situations. As a step towards improving those methods, in this paper we present a new approach for alpha matting. As shown in Fig. 2.1., the proposed method allows a more effective sampling, and thus provides considerably better alpha mattes especially on the object boundaries. Furthermore, we also show that our sampling scheme can be easily extended to video matting by considering optical flow and temporal refinements schemes, resulting in temporally consistent and spatially accurate alpha mattes.

To conclude the introduction, the main contributions of this work can be summarized as follows:
To overcome the limitations of the previous works, we develop a well-founded sampling strategy, which rely on a recently proposed sparse subset selection technique [24], to select a small set of foreground and background samples that best explain the unknown pixels.

We design a new dissimilarity measure between two samples based on KL-divergence between the distributions of the features extracted in the vicinity of the samples. This measure is utilized in both selecting the representative samples and finding the best \((F, B)\) pair for an unknown pixel.

We provide compelling qualitative and quantitative results on a benchmark dataset of images [50] that demonstrate substantial improvements in the estimated alpha mattes upon current state-of-the-art methods.

We extend our sampling strategy to compute alpha mattes for video sequences by incorporating motion information in an effortless way. Besides, we demonstrate the effectiveness of our video matting method on the recently proposed video matting benchmark dataset [51].

We show the feasibility of our method for hard-to-handle case of sparse user inputs on a number of images.

## 2.3. Proposed Approach

In this study, we build upon a recent work by Elhamifar et al. [24], and address the sampling process in image and video matting as a sparse subset selection problem. In particular, we find a few representative pixels for the known foreground and background regions solely based on pairwise dissimilarities between the known and unknown pixels. As in other sampling-based approaches, in our formulation, the dissimilarity measure used in comparing two samples is of great importance since it directly affects the quality of selected samples. As we mentioned earlier, another contribution of this study is a new dissimilarity measure which is based on KL-divergence between feature distributions. In the following, we begin with the definition of our dissimilarity measure, and then discuss the details of the proposed algorithm. The steps of the algorithm involves collecting foreground and background color samples from
known pixels via sparse subset selection, then we define an objective function to find the best \((F, B)\) pair for an unknown pixel according to linear composition equation. After that, we explain how we extend defined sampling strategy to video matting by utilizing temporal information.

### 2.3.1. Dissimilarity Between Two Samples

Sampling-based approaches generally consider very simple measures which depend on chromatic and/or spatial similarities [13, 14, 16, 52]. The only exceptions are [15, 17], which also employ some texture similarity measures. Unlike those measures, here, we consider a statistical data representation and propose to use an information-theoretic approach. In particular, our measure depends on a parametric version of the Kullback-Leibler (KL) Divergence [80], a well-known non-symmetric measure of the difference between two probability distributions in information theory, which we describe below. We note that KL-Divergence was used in a different way for video matting previously in [65].

Given an input image, we extract a 9-dimensional feature vector \(\phi\) for each pixel as follows:

\[
\phi(x, y) = \begin{bmatrix} x & y & r & g & b & |I_x| & |I_y| & |I_{xx}| & |I_{yy}| \end{bmatrix}^T
\]  

(3)

with \((x, y)\) denoting the pixel location, \(I = [r \ g \ b]\) representing the pixel values of the RGB color space, and \(I_x, I_y, I_{xx}, I_{yy}\) respectively corresponding to the first and second-order derivatives of the image intensities, estimated via the filters \([-1 \ 0 \ 1]\) and \([-1 \ 2 \ -1]\) in horizontal and vertical directions.

Next we group the pixels into perceptually meaningful atomic regions using the SLIC algorithm [81]. The motivation behind this step is two folds. First, we use mean color of each foreground or background superpixel to reduce the sample space over which the representative samples are determined. Second, extracting these superpixels helps us to describe a pixel by means of the characteristics of its neighboring pixels, which provides a source of contextual information.
Let \( s_p \) and \( s_q \) respectively denote two superpixels. Then, one can use the KL-divergence to measure the distance between \( s_p \) and \( s_q \) by considering the corresponding feature distributions \( P \) and \( Q \) as

\[
D_{KL}(P||Q) = \int_{-\infty}^{\infty} p(x) \ln \frac{p(x)}{q(x)} \, dx
\]  

(4)

In our formulation, we assume that each feature distribution can be modeled through a multivariate normal distribution such that \( P \sim \mathcal{N}_{s_p} = \mathcal{N} \left( \mu_p, \Sigma_p \right) \). Here, \( p(x) \) and \( q(x) \) respectively denote these probability density functions of \( P \) and \( Q \). Then, the KL-Divergence between two superpixels \( s_p \) and \( s_q \) is described as follows:

\[
D_{KL}(\mathcal{N}_{s_p}||\mathcal{N}_{s_q}) = \frac{1}{2} \left( \text{tr} \left( \Sigma_p^{-1} \Sigma_q \right) + \ln \frac{\det \Sigma_q}{\det \Sigma_p} \right)
\]

\[
+ \left( \mu_q - \mu_p \right)^\top \Sigma_p^{-1} \left( \mu_q - \mu_p \right) - k
\]  

(5)

with \( k = 9 \) denoting our feature dimension.

Note that the KL-divergence is not symmetric, hence we symmetrize it as follows to obtain a distance metric:

\[
\text{dist}(s_p, s_q) = D_{KL}(\mathcal{N}_{s_p}||\mathcal{N}_{s_q}) + D_{KL}(\mathcal{N}_{s_q}||\mathcal{N}_{s_p})
\]  

(6)

In measuring the dissimilarity between two superpixels \( s_p \) and \( s_q \), we found that, instead of using the metric in Eq. (6), the dissimilarity measure derived below lead to better discrimination:

\[
S(s_p, s_q) = \frac{1}{\text{dist}(s_p, s_q) + \epsilon}
\]  

(7)

\[
d(s_p, s_q) = 1 - \min(S(s_p, s_q), 1)
\]  

(8)

where we take \( \epsilon = 0.5 \) in the experiments.
In Figure 2.2, we qualitatively verify the effectiveness of our statistical dissimilarity measure over using only the mean color values of the superpixels. For a given input image, we compute the pairwise dissimilarities between the superpixels extracted from the known foreground and background, and unknown regions and then these values are projected to a 2-dimensional space using t-SNE [53]. As can be seen, the proposed KL-divergence based dissimilarity measure provides better discrimination than simply using color distance.

![Distance embedding visualizations using t-SNE method](image)

Figure 2.2. Distance embedding visualizations using t-SNE method [53] clearly demonstrate that the proposed KL-divergence based dissimilarity measure provides a better discrimination between known foreground and background pixels than using the standard color distance.

### 2.3.2. Sampling via Sparse Subset Selection

Our strategy to obtain representative samples of known foreground and background regions to encode unknown region is inspired by the recently proposed Dissimilarity-based Sparse Subset Selection (DS3) algorithm [24], which formulate subset selection as a row-sparsity regularized trace minimization problem and presents a convex optimization framework to solve it. Suppose we use $K$ and $U$ to represent the set of superpixels extracted from the known foreground ($f$) and background ($b$), and unknown ($u$) regions, with $N = N_f + N_b$ and $M$ elements, respectively:
\[ K = \{ s_1^f, \ldots, s_{N_f}^f, s_1^b, \ldots, s_{N_b}^b \} \]

\[ U = \{ s_1^u, \ldots, s_M^u \} \]  \hspace{1cm} (9)

Assume that the pairwise dissimilarities \( \{d_{ij}\}_{i=1}^{N_f}_{j=1}^{M} \) between superpixels of known region \( K \) and unknown region \( U \) are computed using the dissimilarity measure defined in Eq. (8)\(^2\), and arranged into a matrix form as

\[
D = \begin{bmatrix}
  d_1^T \\
  \vdots \\
  d_N^T
\end{bmatrix} = \begin{bmatrix}
  d_{11} & d_{12} & \cdots & d_{1M} \\
  \vdots & \vdots & \ddots & \vdots \\
  d_{N1} & d_{N2} & \cdots & d_{NM}
\end{bmatrix} \in \mathbb{R}^{N \times M}  \hspace{1cm} (10)
\]

where the entries \( d_{ij} \) signifies how well the superpixel \( i \) represents the superpixel \( j \), the smaller the value, the higher the degree of representativeness.

According to the method described in [24], in order to find a sparse set of samples of \( K \) that well represents \( U \), one can introduce a matrix of variables \( P \in \mathbb{R}^{N \times M} \) as

\[
P = \begin{bmatrix}
  p_1^T \\
  \vdots \\
  p_N^T
\end{bmatrix} = \begin{bmatrix}
  p_{11} & p_{12} & \cdots & p_{1M} \\
  \vdots & \vdots & \ddots & \vdots \\
  p_{N1} & d_{N2} & \cdots & p_{NM}
\end{bmatrix}  \hspace{1cm} (11)
\]

whose each entry \( p_{ij} \in [0, 1] \) is associated to \( d_{ij} \) and denote the probability of superpixel \( i \) being a representative for superpixel \( j \). Then, the problem can be formulated as the following trace minimization problem regularized by a row-sparsity term:

\[
\min_P \quad \gamma \|P\|_{1,\infty} + \text{tr}(D^TP) \\
\text{s.t.} \quad 1^TP = 1^T, P \geq 0
\]  \hspace{1cm} (12)

\(^2\)We note that the approach is quite general in that it could work with dissimilarities which are asymmetric or violate the triangle inequality.
where the first term $\|P\|_{1,\infty} \triangleq \sum_i \|p_i\|_{\infty}$ penalizes the size of the representative set, the second term $\text{tr}(D^TP) = \sum_{ij} d_{ij}p_{ij}$ simply measures the total encoding cost, and the parameter $\gamma$ provides a trade-off between number of samples and encoding quality where smaller values of $\gamma$ will lead to more number of representative samples. An optimal solution $P^*$ can be found very efficiently using an Alternating Direction Method of Multipliers (ADMM) approach [24], in which the indices from the nonzero rows of the solution $P^*$ give us the selected samples of foreground and background superpixels, where we use the mean colors of these superpixels as the candidate set of foreground $F$ and background $B$ colors.

Figure 2.3. shows the samples obtained with our sparse sampling strategy on an illustrative image. As it can be seen, the proposed approach allows robust selection of a small set samples from the known regions where the selected samples are the samples amongst the ones that best represent the unknown regions. Hence, as compared to the existing sampling based models, we employ less number of samples to determine the alpha matte values of the unknown pixels.

### 2.3.3. Selecting The Best \((F, B)\) Pair

As compared to local sampling methods for image matting, which only collect samples near a given unknown pixel, employing a global scheme, such as ours, has the advantage of not missing any true samples if they are not located in the vicinity of the unknown pixel. In some cases, however, there is also a possibility that a local analysis may work better, especially
when local samples are more strongly correlated with the unknown pixel. Hence, to get the best of both worlds, we decide to combine our global sparse sampling strategy with a local sampling scheme. Specifically, for a given unknown pixel, we enlarge the global candidate set to include 10 additional foreground and background samples which are selected from the spatially nearest boundary superpixels.

Once candidate foreground and background colors are sampled for an unknown pixel, we select the best foreground and background pair \((F, B)\) and accordingly determine its alpha matte value. In order to identify the best pair, we define a goodness function that depends on four different measures, which are described in detail below. In particular, in our formulation, we adopt the previously suggested chromatic distortion \(C_u\) and spatial distance \(S_u\) measures \([14–16, 18]\) and additionally propose two new contextual similarity measures \(T_u\) and \(R_u\) to better deal with color ambiguity.

For an unknown pixel \(u\) and a foreground-background pair \((F_i, B_i)\), the chromatic distortion \(C_u\) measures how well the alpha matte \(\hat{\alpha}\) estimated via Eq. (2) from \((F_i, B_i)\) fit to the linear composite equation given by Eq. (1), and is defined as

\[
C_u(F_i, B_i) = \exp\left(-\|I_u - (\hat{\alpha}F_i + (1 - \hat{\alpha})B_i)\|\right)
\]  

(13)

where \(I_u\) denote the observed color of the unknown pixel \(u\).

The spatial distance measure \(S_u\) quantifies the spatial closeness of the unknown pixel \(u\) to the sample pair \((F_i, B_i)\) according to the distance between the coordinates of these pixels. Therefore, it favors selecting samples that are spatially close to the unknown pixel. It is simply defined as

\[
S_u(F_i, B_i) = \exp\left(-\frac{\|u - f_i\|}{Z_F}\right) \cdot \exp\left(-\frac{\|u - b_i\|}{Z_B}\right)
\]  

(14)

where \(f_i\) and \(b_i\) respectively denote the spatial coordinates of the centers of the superpixels that are associated with the foreground and the background samples \(F_i\) and \(B_i\). The scalars \(Z_F = (1/n_F) \sum_{k=1}^{n_F} \|u - f_k\|\) and \(Z_B = (1/n_B) \sum_{k=1}^{n_B} \|u - b_k\|\) are used as scaling factors,
which correspond to the mean spatial distance from the unknown pixel \( u \) to all foreground samples \( F \) with \( n_F \) elements and all background samples \( B \) with \( n_B \) elements, respectively.

One of the great challenges in image matting is the color ambiguity problem which arises when the foreground and background have similar colors. As most of the matting studies consider pixel based similarities in comparing samples, they generally fail to resolve this ambiguity and incorrectly recognize an unknown foreground pixel as background or vice versa. To account for this, we introduce the following two additional local contextual similarity measures \( T_u \) and \( R_u \), which both exploit the similarity function defined in Eq. (7).

The first measure \( T_u \) specifies the compatibility of the unknown pixel with the selected foreground and background samples, computed by means of their statistical feature similarities, and it provides a bias towards those pairs \((F_i, B_i)\) that have local contexts similar to that of the unknown pixel, and is formulated as

$$T_u(F_i, B_i) = S(s_{F_i}, s_u) + S(s_{B_i}, s_u)$$

(15)

where \( s_{F_i}, s_{B_i}, \) and \( s_u \) respectively denote the superpixels associated with the corresponding foreground and background samples and the unknown pixel.

The second measure \( R_u \) corresponds to a variant of the robustness term in [52], which builds upon the assumption that for any mixed pixel whose color is affected by both the foreground and the background, the true background and foreground colors have similar feature statistics, calculated over the corresponding superpixels. Thus, it favors the selection of the foreground and the background samples that have similar contexts, and is defined as

$$R_u(F_i, B_i) = S(s_{F_i}, s_{B_i}).$$

(16)
Putting these four measures together, we arrive at the following objective function to determine the best \((F, B)\) pair:

\[
O_u(F_i, B_i) = C_u(F_i, B_i)^c \cdot S_u(F_i, B_i)^s \cdot T_u(F_i, B_i)^t \cdot R_u(F_i, B_i)^r,
\]  
(17)

where \(c, s, t, r\) are weighting coefficients, representing the contribution of the corresponding terms to the objective function. Empirically, we observed that the color distortion \(C_u\) and the contextual similarity measure \(T_u\) are more distinguishing than others, and thus we set the coefficients as \(c = 2, s = 0.5, t = 1, r = 0.5\). Brute-force optimization is done on the objective function in Eq.(17) to select the best background and foreground color samples.

### 2.3.4. Pre- and Post-Processing

Motivated by recent sampling based matting studies \([16, 18]\), we apply some pre- and post-processing steps. First, before selecting the best \((F, B)\) sample pairs, we expand known regions to unknown regions by adopting the pre-processing step used in \([16, 18]\). Specifically, we consider an unknown pixel \(u\) as a foreground pixel if the following condition is satisfied for a foreground pixel \(f\):

\[
(D(I_u, I_f) < E_{thr}) \land (\|I_u - I_f\| \leq (C_{thr} - D(I_u, I_f))),
\]  
(18)

where \(D(I_u, I_f)\) and \(\|I_u - I_f\|\) are the spatial and the chromatic distances between the pixels \(u\) and \(f\), respectively, and \(E_{thr}\) and \(C_{thr}\) are the corresponding thresholds which are all empirically set to 9. Similarly, an unknown pixel \(u\) is taken as a background pixel if a similar condition is met for a background pixel \(b \in B\).

Second, as a post-processing, we perform smoothing on the estimated alpha matte by adopting a modified version of the Laplacian matting model \([5]\) as suggested in \([13]\). That is, we determine the final alpha values \(\alpha^*\) by solving the following global minimization
\[
\alpha^* = \arg\min_{\alpha} \alpha^\top L \alpha + \lambda (\alpha - \hat{\alpha})^\top \Lambda (\alpha - \hat{\alpha}) \\
+ \delta (\alpha - \hat{\alpha})^\top \Delta (\alpha - \hat{\alpha})
\] (19)

where the data term imposes the final alpha matte to be close to the estimated alpha matte \(\hat{\alpha}\) from Eq. (2), and the matting Laplacian \(L\) enforces local smoothing. The diagonal matrix \(\Lambda\) in the first data term is defined using the provided trimap such that it has values 1 for the known pixels and 0 for the unknown ones. The scalar \(\lambda\) is set to 100 so that it ensures no smoothing is applied to the alpha values of the known pixels. The second diagonal matrix \(\Delta\), on the other hand, is defined by further considering the estimated confidence scores in a way that it has values 0 for the known pixels and the corresponding confidence values \(O_{u}(F, B)\) from Eq. (17) for the unknown pixels. The scalar \(\delta\) here is set to 0.1 and determines the relative importance of the smoothness term which considers the correlation between neighboring pixels.

2.4. Extension To Video Matting

As discussed in the previous section, a successful color sampling method should overcome the color ambiguity problem which occurs when the foreground and the background have similar color distributions. Fortunately, in video matting, the temporal motion information in the video sequences provides extra information to disambiguate this problem in the presence of dissimilar motion patterns. In this regard, we extend proposed sampling strategy to exploit temporal information by addressing both the missing true samples and the color ambiguity problems to obtain more accurate alpha mattes.

2.4.1. Motion Aware Temporal Sampling

The similarity metric employed in our sampling approach is quite generic in that we can incorporate any visual feature including motion information in a fairly straightforward way.
In our case, we extend the feature vector $\phi$ in Eq. (3) with the optical flow vectors obtained by [82], as follows:

$$\phi(x, y) = \left[ x \ y \ r \ g \ b \ |I_x| \ |I_y| \ |I_{xx}| \ |I_{yy}| \ v_x \ v_y \right]^T$$

where $(v_x, v_y)$ are optical flow vectors. By doing so, we let the proposed KL-divergence based dissimilarity measure consider contextual motion along with color and orientation.

Such a feature distribution in a local region allows the proposed dissimilarity measure given in Eq. (5) to discriminate color, texture and motion information within our sparse subset selection phase (Eq. (12)). We further use the objective function given in Eq. (17) to select the best $(F, B)$ pair for each unknown pixel. This objective function is also used as a confidence value in the alpha refinement step that will be explained in the following section. As a result, motion information is involved in each step of our video matting method.

Sampling from a single frame can be insufficient due to various types of changes between the video frames such as changes in the illumination, occlusion and changing topology. For this reason, we expand the sampling space in Eq. (9) by including the known foreground and background superpixels from both the previous frame $t - 1$ and the current frame $t$ as:

$$K = \{s_f^{t-1}, \ldots, s_f^{t-1}, s_b^{t-1}, \ldots, s_b^{t-1}, s_f^{t}, \ldots, s_f^{t}, s_b^{t}, \ldots, s_b^{t}\}$$

$$U = \{s_u^{t}, \ldots, s_u^{t}\}$$

This definition extends the dissimilarity matrix $D$ between the unknown and known superpixels in Eq. (10) to include the elements $\{d_{ij}\}_{i=1,\ldots,M}$, where $M$ is the number of unknown superpixels and $N = N_f^{t-1} + N_b^{t-1} + N_f^t + N_b^t$ is the number of known foreground and background superpixels extracted from the previous frame $t - 1$ and the current frame $t$. After constructing the dissimilarity matrix and related probability matrix $P$ in Eq. (11), we solve Eq. (12) to pick up the representative superpixels and accordingly the color samples.

Similar to image matting, we enrich global samples with local samples from the current frame. We apply the same procedure that we used in image matting to select the best $(F, B)$
pair but note that the motion information is incorporated into the feature vectors so that the objective function now contains temporal information via our KL-divergence based dissimilarity measure. Figure 2.4. shows the temporal samples obtained by our modified sampling strategy. The representative samples for the brown background are chosen from the previous frame as the corresponding region becomes occluded by the foreground object in the current frame.

2.4.2. Alpha Refinement

After the alpha matte is estimated based on the selected color samples using Eq. 2, we further refine it by post-processing. For this purpose, some video matting methods [64, 66, 68] employ a 3D matting Laplacian defined over a multi-frame neighborhood by warping the neighboring frames to current frame via optical flow. It provides a better temporal coherency as compared to the standard matting Laplacian [5]. However, we observe that obtaining better temporal coherences might worsen the spatial accuracy due to inaccurate estimation of optical flows. Hence, in our experiments, we only consider the standard matting Laplacian that we extend with additional motion confidences.

\[
\alpha^t = \arg\min_{\alpha^t} \alpha^T L^t \alpha^t + \lambda (\alpha^t - \hat{\alpha}^t)^T \Lambda^t (\alpha^t - \hat{\alpha}^t) \\
+ \delta (\alpha^t - \hat{\alpha}^t)^T \Delta^t (\alpha^t - \hat{\alpha}^t) 
\]  

(22)

where \(L^t\) is the matting Laplacian, and the other terms are the data fidelity terms. The difference with Eq. (19) lies in the diagonal matrix \(\Delta^t\). Specifically, for each unknown pixel, it has the confidence value \(O_u^t(F^t, B^t)\) estimated from the modified motion-aware sampling scheme, and 0 for the remaining known pixels at the frame \(t\).
2.5. Experiments

We evaluate our alpha matting approach on benchmark datasets used for evaluating image [50] and video[51] matting. First, we conduct an extensive evaluation of the proposed sampling strategy for image matting by providing qualitative and quantitative results and investigate the effects of the parameters in detail. Second, we compare our extension to video matting against state-of-the-art video matting methods both qualitatively and in terms of a set of spatial and temporal quality metrics. Next, we demonstrate that our sampling strategy can also cope with sparse user inputs, beating some recent methods especially proposed for this kind of input. Lastly, we argue the computational complexity of proposed approach.

2.5.1. Image Matting

Image matting benchmark dataset [50] contains 35 natural images and each image has a foreground object with different degrees of translucency or transparency. Among those images, 27 of them constitute the training set where the groundtruth alpha mattes are available. On the other hand, the remaining 8 images are used for the actual evaluation, whose ground truth alpha mattes are hidden from the public to prevent parameter tuning. In addition, for each test image, there are three matting difficulty levels that respectively correspond to small, large and user trimaps. To quantitatively evaluate our approach, in the experiments, we consider three different metrics, namely, the mean square error (MSE), the
Figure 2.5. Effect of $\gamma$ parameter on the performance. Plot shows average MSE values over all training images and all trimaps.

sum of absolute differences (SAD) and the gradient error. We do not report connectivity scores as it is argued in [50] that it is not a robust measure.

**Effect of $\gamma$ parameter.** Fig. 2.5. shows that the average MSE values over all the training images and all trimaps do not vary much for different values of $\gamma$. These results seem to be consistent with the theoretical analysis in [24] that for a proper range of values, the DS3 algorithm that we utilize in sampling is guaranteed to find representative samples from all groups when there is a mutual relationship between known and unknown sets. In the remaining experiments, $\gamma$ is set to 0.025 as it provides the minimum MSE value for the training set.

**Effect of local samples.** In Fig. 2.6., we show the effect of including local samples from boundary to the candidate set found by the proposed sparse sampling scheme. The numbers in the plot refer to the MSE errors averaged over all test images. For each trimap type, adding some closest boundary pixels further improves the performance. The smallest gain is in the large trimaps since having more number of known pixels helps our sparse sampling method to better exploit the associations between the known and unknown regions, eliminating the need for local samples.
Figure 2.6. Effect of including local samples to the representative set obtained with the proposed sparse sampling scheme. Plot shows average MSE values over all test images for three types of trimaps.

Figure 2.7. Visual comparison of our approach with other sampling-based image matting methods. (a) Input image, (b) CWCT sampling [17], (c) Comprehensive sampling [16], (d) LNSP matting [54], (e) Sparse coded matting [18] and (f) Proposed approach.

**Comparison with the state-of-the-art.** Table 2.1. presents the quantitative comparison of our approach and nine best performing matting algorithms on the benchmark hosted at www.alphamatting.com [50] where we report the average rankings over the test images according to SAD, MSE and gradient metrics for all three different types of trimap, and the
Table 2.1. Evaluation of matting methods on the benchmark dataset [50] with three trimaps according to SAD, MSE and Gradient error.

<table>
<thead>
<tr>
<th>Method</th>
<th>Sum of Absolute Differences</th>
<th>Mean Square Error</th>
<th>Gradient Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>overall rank</td>
<td>avg.</td>
<td>avg. large rank</td>
</tr>
<tr>
<td>DCNN Matting</td>
<td>2.4</td>
<td>3.4</td>
<td>1.3</td>
</tr>
<tr>
<td>CSC Matting</td>
<td>9.1</td>
<td>13</td>
<td>9.1</td>
</tr>
<tr>
<td>Graph-based sparse matting</td>
<td>9.6</td>
<td>9.9</td>
<td>10</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>9.8</td>
<td>7.4</td>
<td>9.5</td>
</tr>
<tr>
<td>TIPS-RV Matting</td>
<td>11.1</td>
<td>9.6</td>
<td>10.1</td>
</tr>
<tr>
<td>Iterative Transductive Matting</td>
<td>11.8</td>
<td>12.8</td>
<td>11.5</td>
</tr>
<tr>
<td>Compressed sampling</td>
<td>14.4</td>
<td>8.1</td>
<td>12</td>
</tr>
<tr>
<td>SVR Matting</td>
<td>12.3</td>
<td>14.8</td>
<td>11.8</td>
</tr>
<tr>
<td>CW Color and Texture</td>
<td>12.3</td>
<td>12.4</td>
<td>15</td>
</tr>
</tbody>
</table>

Overall ranks, computed as the average over all the images and for all the trimaps. Overall, our approach provides highly competitive results against the state-of-the-art methods especially the ones over sampling-based approaches that do not consider deep learning. It ranked the third best with respect to the gradient error and mean square error and the fifth best for the sum of absolute differences. Especially, it outperforms all the existing sampling-based matting methods. Fig. 2.7. provides qualitative comparisons of our approach and the recent matting studies [16–18, 54] on the doll, troll and net images from the benchmark dataset. In Table 2.2., we also present comparison with recent related methods which were appeared after publication of our method. As can be seen, our method still produces highly competitive results.

Textured background. In the first row of Fig. 2.7., we show the ability of our approach to naturally handle textured backgrounds via the proposed KL-divergence based contextual measure. For the doll placed in front of a highly textured background, while other matting methods, including CWCT sampling [17] which employs an additional texture compatibility measure, tend to interpret some of the colored blobs in the background as foreground, our model produces a much more accurate alpha map.
### Table 2.2. Evaluation of recent related matting methods on the benchmark dataset [50] with three trimaps according to SAD and MSE error.

<table>
<thead>
<tr>
<th>Method</th>
<th>Sum of Absolute Differences</th>
<th>Mean Square Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>over-</td>
<td>avg.</td>
</tr>
<tr>
<td></td>
<td>all</td>
<td>small</td>
</tr>
<tr>
<td>1. SampleNet [23]</td>
<td>3.5</td>
<td>2.5</td>
</tr>
<tr>
<td>3. IFM [10]</td>
<td>6.1</td>
<td>7.4</td>
</tr>
<tr>
<td>5. Proposed Method</td>
<td>18.5</td>
<td>17.5</td>
</tr>
</tbody>
</table>

* We have received the up-to-date results from SampleNet matting [23].

**Color ambiguity.** When the foreground object and the background have similar color distributions, most matting studies suffer from the so-called color ambiguity and fail to provide reliable alpha values for the unknown pixels. Both second and third rows of Fig. 2.7. illustrate this issue where the colors of the book and the bridge in the background is very similar to those of the hairs of the doll and troll, respectively. For these examples, CWCT [17] and Comprehensive sampling [16] give inaccurate estimations whereas LNSP matting [54] oversmooths the foreground matte. Sparse coded matting [18] provides better results but misses some of the foreground details in the hairs. On the other hand, our method is able to achieve significantly better results, providing a more robust discrimination between the background and the foreground.

**Missing samples.** Previously proposed sampling-based matting methods typically employ certain assumptions while collecting samples from known regions but these assumptions might sometimes lead to missing true foreground and background colors for some unknown pixels. In the fourth row of Fig. 2.7., we demonstrate the effectiveness of our sparse sampling strategy on the troll image. While the other sampling based methods [16–18] incorrectly recognize the blue ribbon as mixed pixels, our algorithm successfully interprets it as a part of the foreground object. Likewise, the LNSP matting [54] produces an alpha map similar to ours as it uses a non-local smoothness prior in their formulation. If known regions do not include some color samples representing the colors from the unknown region, our method might not give accurate alpha matte results as seen for the plastic bag image.
Translucent foreground. Transparent or translucent objects pose another great challenge for matting as they make collecting true foreground color samples difficult. The last two rows of Fig. 2.7. show the results of two different regions from the net image in detail where such a foreground object exists. Due to the characteristics of the test image, all of the competing matting methods fail to differentiate background pixels from the foreground although the distributions of the background and foreground colors are well separated. In contrast, our approach produces a remarkably superior alpha matte.

2.5.2. Video Matting

For our video matting experiments, we use the very recently proposed video matting benchmark dataset [51] (www.videomatting.com) to evaluate our video matting results. This dataset includes 3 training sequences with available ground-truth maps, and 10 testing sequences with hidden ground-truth maps. Furthermore, for each video frame, 3 different trimaps are generated according to the size of the unknown region as narrow, medium and wide.

Effect of temporal sampling and motion features. Almost all video matting methods [60–62, 65–68] employ optical flow fields as motion features. However, as discussed in the related papers, optical flow estimation is not always perfect, which may deterioate the quality of extracted alpha mattes. We analyze the effect of our optical-flow based motion features on the training sequences of [51] (Alex, Castle, Dmitriv, see Fig. 2.8.), where the ground truth alpha mattes are available. In Table 2.3., we present the effects of motion and temporal sampling in terms of SSD (sum of squared distances) metric. As can be seen, motion and temporal sampling improve the alpha matte results for Alex and Dmitriv sequences, however for castle sequence worsen the results. The Castle sequence has a complex foreground object and a highly textured background region as compared to other sequences, and we think that these factors negatively affect the extracted optical flow vectors, and consequently the quality of our temporal sampling. In the remaining experiments, on the other hand, we report our results
Table 2.3. Effect of spatial sampling (single frame) vs. temporal sampling (two frames) and optical flow information (OF) to alpha matte results on SSD scores

<table>
<thead>
<tr>
<th>Video</th>
<th>Trimap</th>
<th>Spatial</th>
<th>Spatial+OF</th>
<th>Temporal+OF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alex</td>
<td>Narrow</td>
<td>2.134</td>
<td>2.129</td>
<td>2.125</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>1.740</td>
<td>1.707</td>
<td>1.689</td>
</tr>
<tr>
<td></td>
<td>Wide</td>
<td>1.852</td>
<td>1.753</td>
<td>1.729</td>
</tr>
<tr>
<td>Castle</td>
<td>Narrow</td>
<td>6.736</td>
<td>7.098</td>
<td>7.158</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>7.160</td>
<td>7.591</td>
<td>7.624</td>
</tr>
<tr>
<td></td>
<td>Wide</td>
<td>7.671</td>
<td>8.150</td>
<td>8.208</td>
</tr>
<tr>
<td>Dmitriv</td>
<td>Narrow</td>
<td>1.918</td>
<td>1.916</td>
<td>1.917</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>2.409</td>
<td>2.336</td>
<td>2.329</td>
</tr>
<tr>
<td></td>
<td>Wide</td>
<td>2.734</td>
<td>2.662</td>
<td>2.652</td>
</tr>
</tbody>
</table>

with both using motion features and temporal sampling since this setup provides the best results on the overall training sequences.

Figure 2.8. Training sequences from the video matting benchmark dataset [51].

**Comparison with other methods.** Table 2.4. shows the quantitative evaluation of different matting methods on the video matting benchmark dataset [51]. Evaluation is carried out on the test sequences according to the quality metrics which highlight spatial accuracy and temporal coherency of the estimated alpha mattes. Specifically, SSDA (Sum of Squared Distances) error measure is used to evaluate the accuracy of the estimated alpha matte for each pixel. Two additional temporal-coherency metrics, which measure deterioration ratio of the alpha mattes over consequent frames, are used to test the temporal coherency. SSDdt measures the overall variation in the sum squared distances between the estimated alpha matte and the ground-truth for each pixel over the consecutive frames. MESSDdt measure is the generalized version of SSDdt, which additionally considers the optical flow information over video frames. Thus, it provides a more robust comparison of specifically the motion-aware matting methods.
Table 2.4. Evaluation of matting methods on the benchmark dataset [51] with three trimaps according to SSDA, SSDdt and MESSDdt.

<table>
<thead>
<tr>
<th>Method</th>
<th>SSDA</th>
<th></th>
<th></th>
<th></th>
<th>Method</th>
<th>SSDA</th>
<th></th>
<th></th>
<th></th>
<th>Method</th>
<th>SSDA</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>over-</td>
<td>avg.</td>
<td>avg.</td>
<td>avg.</td>
<td>over-</td>
<td>avg.</td>
<td>avg.</td>
<td>avg.</td>
<td>avg.</td>
<td>over-</td>
<td>avg.</td>
<td>avg.</td>
<td>avg.</td>
<td>avg.</td>
</tr>
<tr>
<td></td>
<td>rank</td>
<td>narrow</td>
<td>medium</td>
<td>wide</td>
<td>rank</td>
<td>narrow</td>
<td>medium</td>
<td>wide</td>
<td>rank</td>
<td>narrow</td>
<td>medium</td>
<td>wide</td>
<td>rank</td>
<td></td>
</tr>
<tr>
<td>1. Proposed Method</td>
<td>2.3</td>
<td>2.2</td>
<td>2.1</td>
<td>1.7</td>
<td>1. Learning Based</td>
<td>2.6</td>
<td>2.5</td>
<td>2.5</td>
<td>2.7</td>
<td>1. Learning Based</td>
<td>2.4</td>
<td>2.3</td>
<td>2.3</td>
<td>2.4</td>
</tr>
<tr>
<td>2. Learning Based</td>
<td>2.7</td>
<td>2.5</td>
<td>2.7</td>
<td>3</td>
<td>2. Closed Form</td>
<td>3.1</td>
<td>2.8</td>
<td>3.5</td>
<td>3.1</td>
<td>2. Closed Form</td>
<td>2.7</td>
<td>2.4</td>
<td>2.6</td>
<td>3</td>
</tr>
<tr>
<td>3. Comprehensive</td>
<td>3.6</td>
<td>4</td>
<td>3.3</td>
<td>3.4</td>
<td>3. Refine Edge</td>
<td>3.4</td>
<td>3.9</td>
<td>3.4</td>
<td>2.9</td>
<td>3. Proposed Method</td>
<td>3.6</td>
<td>3.9</td>
<td>3.5</td>
<td>3.4</td>
</tr>
<tr>
<td>Sampling</td>
<td>3.9</td>
<td>3.9</td>
<td>3.7</td>
<td>4.1</td>
<td>4. Proposed Method</td>
<td>4</td>
<td>4.9</td>
<td>3.7</td>
<td>3.4</td>
<td>4. Refine Edge</td>
<td>3.6</td>
<td>4.3</td>
<td>3.7</td>
<td>2.8</td>
</tr>
<tr>
<td>4. Closed Form</td>
<td>5.3</td>
<td>4.9</td>
<td>5.4</td>
<td>5.5</td>
<td>5. Comprehensive Sampling</td>
<td>5</td>
<td>4.8</td>
<td>4.9</td>
<td>5.2</td>
<td>5. Comprehensive Sampling</td>
<td>5</td>
<td>4.9</td>
<td>4.9</td>
<td>5.3</td>
</tr>
<tr>
<td>5. Shared Matting</td>
<td>5.5</td>
<td>4.5</td>
<td>5.6</td>
<td>6.4</td>
<td>6. Robust Matting</td>
<td>5.2</td>
<td>4.3</td>
<td>5.4</td>
<td>6</td>
<td>6. Shared Matting</td>
<td>6.2</td>
<td>5.7</td>
<td>6.5</td>
<td>6.6</td>
</tr>
<tr>
<td>6. Robust Matting</td>
<td>5.5</td>
<td>5.8</td>
<td>5.5</td>
<td>5.2</td>
<td>7. Refine Edge</td>
<td>5.6</td>
<td>5.6</td>
<td>5.9</td>
<td>6</td>
<td>7. Robust Matting</td>
<td>6.4</td>
<td>5.4</td>
<td>6.4</td>
<td>7.3</td>
</tr>
<tr>
<td>7. Refine Edge</td>
<td>8.4</td>
<td>8.9</td>
<td>8.6</td>
<td>7.8</td>
<td>8. KNN Matting</td>
<td>8.5</td>
<td>8.8</td>
<td>8.5</td>
<td>8.2</td>
<td>8. KNN Matting</td>
<td>7.6</td>
<td>8.4</td>
<td>7.6</td>
<td>6.9</td>
</tr>
</tbody>
</table>

Figure 2.9. Visual comparison between the proposed and the other best-performing video matting methods on the video benchmark dataset [51]. (a) Input video frames, (b) Closed Form [5], (c) Refine Edge Tool, (d) Learning Based [19] and (e) Proposed approach.

Matting methods are sorted by the average ranking scores according to the accuracy metric SSDA and the temporal-coherency metrics SSDdt and MESSDdt. As can be seen from Table 2.4., our method gives the best results for the spatial accuracy metric and highly competitive results for the temporal-coherency metrics. Here, it is worth noting that the
motion-aware MESSDdt metric generally provides a more robust comparison of the methods than the SSDdt metric since the inaccurate optical flow estimations could worsen the quality of alpha mattes as we analyzed above.

In Fig. 2.9., we also give the qualitative comparisons of our approach against the other best-performing matting methods [5],[19] and the Refine Edge Tool of Adobe on the Juneau sequence, for the frames 45, 46 and 47. The zoomed regions demonstrate that our method is affected from the missing true color and the color ambiguity far less than the other methods since it inherently employs motion information in sampling and uses a sampling strategy extended to consider temporal information as well.

2.5.3. Sparse Scribbles

Our approach can also work with sparse user inputs since we do not make any spatial assumption while collecting color samples, which is the case for many previous sampling-based image matting methods [13, 14, 16, 52]. More specifically, we apply our sampling strategy on the sparse user inputs by considering the superpixels that contain any user scribbled image pixel as the known superpixels and the others as the unknown superpixels. Consequently, we construct the dissimilarity matrix between known and unknown superpixels via our proposed KL-Divergence based dissimilarity measure and select the representative color samples from the known user scribbles using the sparse subset selection strategy described in Eq. (12).

We compare the performance of our method with KNN Matting [7] and Nonlocal Matting [6] methods which are both tailor-fit to work with sparse user inputs and with Closed Form [5] and Comprehensive sampling [16] methods. In Fig. 2.10., some image matting results along with the estimated MSE (Mean Square Error) scores are given. Our method in general produces better results than all these methods. This also demonstrates that the proposed sampling scheme for image matting is a generic and theoretically well-grounded sampling strategy for alpha matting problem.
2.5.4. Runtime Performance

In our work, we used the ADMM-based serial implementation of the DS3 method, but it is indeed highly parallelizable [24]. Overall, the runtime performance of our current implementation is better than Comprehensive sampling (CS) as our algorithm selects much less and more representative samples from the known regions, which significantly reduces runtime costs of the subsequent steps. For example, for the doll, donkey and elephant images in [50], the average running times over all trimaps are 341 secs for our method, and 414 secs for CS, on a PC with an Intel Xeon 2GHz CPU.

Figure 2.10. Alpha matting results with scribble based user input and the corresponding MSE scores estimated over the input images.
2.6. Discussion

In this section, we developed a new and theoretically well-grounded sampling strategy for image matting and extended it to video matting. Rather than making assumptions about the possible locations of true color samples, or performing a direct clustering of all known pixels, our sampling scheme solves a sparse subset selection problem over known pixels to obtain a small set of representative samples that best explain the unknown pixels. This property also makes our sampling method directly applicable to sparse user inputs provided to estimate alpha matte. Moreover, it employs a novel KL-divergence based contextual measure in both collecting the candidate sample set and finding the best \((F, B)\) pair for an unknown pixel. Our experiments on both image and video benchmark datasets clearly demonstrate that our approach is superior to existing sampling-based image and video matting methods and achieves state-of-the-art results.
3. MANIPULATING ATTRIBUTES OF NATURAL SCENES VIA HALLUCINATION

“The trees, being partly covered with snow, were outlined indistinctly against the grayish background formed by a cloudy sky, barely whitened by the moon.”

– Honore de Balzac (Sarrasine, 1831)

![Figure 3.1. Given a natural image, our approach can hallucinate different versions of the same scene in a wide range of conditions, e.g. night, sunset, winter, spring, rain, fog or even a combination of those. First, we utilize a generator network to imagine the scene with respect to its semantic layout and the desired set of attributes. Then, we directly transfer the scene characteristics from the hallucinated output to the input image, without the need for a reference style image.](image)

The visual world we live in constantly changes its appearance depending on time and seasons. For example, at sunset, the sun gets close to the horizon gives the sky a pleasant red tint, with the advent of warm summer, the green tones on the grass leave its place in bright yellowish tones and autumn brings a variety of shades of brown and yellow to the trees. Such visual changes in the nature continues in various forms at almost any moment with the effect of time, weather and season. Such high-level changes are referred to as transient scene attributes – e.g. cloudy, foggy, night, sunset, winter, summer, to name a few [41].
Recognizing transient attributes of an outdoor image and modifying its content to reflect any changes in these properties were studied in the past, however, current approaches have many constraints which limit their usability and effectiveness in attribute manipulation. In this work, we present a framework that can hallucinate different versions of a natural scene given its semantic layout and its desired real valued transient attributes. Our model can generate many possible output images from scratch such as the ones in Fig. 3.1., which is made possible by learning from data the semantic meaning of each transient attribute and the corresponding local and global transformations.

3.1. Introduction

Image generation is quite a challenging task since it needs to have realistic looking outputs. Visual attribute manipulation can be considered a bit harder as it aims at photorealism as well as results that are semantically consistent with the input image. For example, for predicting the look of a scene at sunset, visual appearances of the sky and the ground undergo changes differently, the sky gets different shades of red while the dominant color of the ground becomes much darker and texture details get lost. Unlike recent image synthesis methods [34, 39, 40, 83], which explore producing realistic-looking images from semantic layouts, automatically manipulating visual attributes requires modifying the appearance of an input image while preserving object-specific semantic details intact. Some recent style transfer methods achieve this goal to a certain extent but they require a reference style image [42, 43].

A simple solution to obtain an automatic style transfer method is to retrieve reference style images with desired attributes from a well-prepared dataset with a rich set of attributes. However, this approach raises new issues that need to be solved such as retrieving images according to desired attributes and semantic layout in an effective way. To overcome these obstacles, we propose to combine neural image synthesis and style transfer approaches to perform visual attribute manipulation. For this purpose, we first devise a conditional image synthesis model that is capable of hallucinating desired attributes on synthetically generated images.

This work has been accepted to ACM Transactions on Graphics for publication.
scenes with semantic content similar to the input image and then we resort to a photo style transfer method to transfer the visual look of the hallucinated image to the original input image to produce a resulting image with the desired attributes.

A rich variety of generative models including Generative Adversarial Networks (GANs) [25–27], Variational Autoencoders (VAEs) [84, 85], and autoregressive models [86, 87] are developed to synthesize visually plausible images. Images of higher resolutions, e.g. 256×256, 512×512 or 1024×1024, have also been rendered under improved versions of these frameworks [28–36]. However, generating diverse, photorealistic and well-controlled images of complex scenes has not yet been fully solved. For image synthesis, we propose a new conditional GAN based approach to generate a target image which has the same semantic layout with the input image but reflects the desired transient attributes. As shown in Fig. 2.1, our approach allows users to manipulate the look the an outdoor scene with respect to a set of transient attributes, owing to a learned manifold of natural images.

To build the aforementioned model, we argue the necessity of better control over the generator network in GAN. We address this issue by conditioning ample concrete information of scene contents to the default GAN framework, deriving our proposed attribute and semantic layout conditioned GAN model. Spatial layout information tells the network where to draw, resulting in clearly-defined object boundaries and transient scene attributes serve to edit visual properties of a given scene so that we can hallucinate desired attributes for input image in semantically similar generated image.

However, naively importing the side information is insufficient. For one, when training the discriminator to distinguish mismatched image-condition pairs, if the condition is randomly sampled, it can easily be too off in describing the image to provide meaningful error derivatives. To address this issue, we propose to selectively sample mismatched layouts for a given real image, inspired by the practice of hard negative mining [88]. For another, given the challenging nature of the scene generation problem, adversarial objective alone can struggle to discover a satisfying output distribution. Existing works in synthesizing complex images apply the technique of “feature matching”, or perceptual loss [34, 89]. Here, we also
adopt perceptual loss to stabilize and improve adversarial training for more photographic
generation but contrasting prior works, our approach employs the layout-invariant features
pretrained on segmentation task to ensure consistent layouts between synthesized images
and reference images. For photo style transfer, we use a recent deep learning based approach
[43] which transfers visual appearance between same semantic objects in real photos using
semantic layout maps.

Our contributions are summarized as follows:

- We propose a new two-stage visual attribute manipulation framework for changing
  high-level attributes of a given outdoor image.
- We develop a conditional GAN variant for generating natural scenes faithful to given
  semantic layouts and transient attributes.
- We build up an outdoor scene dataset annotated with layout and transient attribute
  labels by combining and annotating images from Transient Attributes [41] and
  ADE20K [55].

Our code and models are publicly available at the project website[4].

3.2. Related Work

3.2.1. Image Synthesis

In the past few years, much progress has been made towards realistic image synthesis; in
particularly, different flavors and improved versions of Generative Adversarial Networks
(GANs) [25] have achieved impressive results along this direction. Radford et al. [26] were
the first to propose a architecture that can be trained on large scale datasets, which sparked
a wave of studies aimed at improving this line of work [90–92]. Larsen et al. [93] integrates
adversarial discriminator to VAE framework in an attempt to prevent mode collapsing. Its

extension [30] further tackles this issue while improving generation quality and resolution. More recently, Karras et al. [36] have suggested to use a cascaded set of generators to increase both the photorealism and the resolution of generated images. In the subsequent work, Karras et al. [35] have achieved further improvement in realism and diversity of the generated synthetic images by adopting ideas from style transfer literature [94].

Conditional GANs (CGANs) [37] that leverages side information have been widely adopted to generate images under predefined constraints. For example, the recently proposed BigGAN [95] generates high quality, high resolution images conditioned on visual classes in ImageNet. Reed et al. [28, 29] generate images using natural language descriptions; Antipov et al. [96] follow similar pipelines to edit a given facial appearance based on age. Pix2pix [39] undertakes a different approach to conditional generation that it directly translates one type of image information to another type through an encoder-decoder architecture coupled with adversarial loss; its extension Cycle-GAN [33] conducts similar translation under the assumption that well-aligned image pairs are not available. The design of our image synthesis model resembles CGANs, as opposed to Pix2pix, since those so-called image-to-image translation models are limited in terms of output diversity.

In the domain of scene generation, the aforementioned Pix2pix [39] and Cycle-GAN [33] both manage to translate realistic scene images from semantic layouts. However, these models are deterministic, in other words, they can only map one input image to one output image in different domains. Recently, some researchers have proposed multimodal (e.g. BicycleGAN [97]) or multi-domain (e.g. StarGAN [98], MUNIT [99]) image-to-image translation models. Both of these approaches have the ability to translate a given input image to multiple possible output images with the use of a single network. However, in BicycleGAN, the users have no control over the generation process other than deciding upon the source and target domains. StarGAN and MUNIT can perform many-to-many translations but these the translations are always carried out between two different modalities. Although these works improve the diversity to a certain degree, they are still limited in the sense that they do not allow to fully control the latent scene characteristics. For instance, these methods can not generate an image with a little bit of sunset and partly cloudy skies
from an image taken on a clear day. Our proposed model, on the other hand, allows the users to play with all of the scene attributes with varying degrees of freedom at the same time.

Alternatively, some efforts on image-to-image translation has been made to increase the realism and resolution with multi-scale approaches [34, 40, 83]. Wang et al. [40]’s Pix2pixHD model improves both the resolution and the photorealism of Pix2pix [39] by employing multi-scale generator and discriminator networks. Qi et al. [83] utilize a semi-parametric approach and increase the photorealism of the output images obtained from semantic layouts by composing real object segments from a set of training images within an image-to-image synthesis network. Chen et al. [34] try to achieve realistic looking scenes through a carefully crafted regression objective that maps a single input layout to multiple potential scene outputs. Nonetheless, despite modeling one-to-many relationships, the number of outputs is pre-defined and fixed, which still puts tight constraints on the generation process. As compared to these works, besides taking semantic layout as input, our proposed scene generation network is additionally aware of the transient attributes and the latent random noises characterizing intrinsic properties of the generated outputs. As a consequence, our model is more flexible in generating the same scene content under different conditions such as lighting, weather, and seasons.

From training point of view, a careful selection of “negative” pairs, i.e. negative mining, is an essential component in metric learning and ranking [100–102]. Existing works in CGAN have been using randomly sampled negative image-condition pairs [29]. However, such random negative mining strategy has been shown to be inferior to more meticulous negative sampling schemes [103]. Particularly, the negative pair sampling scheme proposed in our work is inspired by the concept of relevant negative [102], where the negative examples that are visually similar to positive ones are emphasized more during learning.

To make the generated images look more similar to the reference images, a common technique is to consider feature matching which is commonly employed through a perceptual loss [34, 89, 104]. The perceptual loss in our proposed model distinguishes itself from
existing works by matching segmentation invariant features from pre-trained segmentation networks [55], leading to diverse generations that comply with the given layouts.

3.2.2. Image Editing

There has been a great effort towards building methods for manipulating visual appearance of a given image. Example-based approaches [105, 106] use a reference image to transfer color space statistics to input image so that visual appearance of input image looks like the reference image. In contrast to these global color transfer approaches, which require highly consistent reference images with input image, user controllable color transfer techniques were also proposed [107, 108] to consider spatial layouts of input and reference images. Dale et al. [108] search for some reference images which have similar visual context to input image in a large image dataset to transfer local color from them and then use color transferred image to restore input image. Other local color transfer approaches [109] use the semantic segments to transfer color between regions in reference and input images have same semantic label (e.g. color is transferred from sky region in reference image to sky region in input image). Some data-driven approaches [41, 110] leverage the time-lapse video datasets taken for same scene to capture scene variations that occur at different times. Shih et al. [110] aim to give times of day appearances to a given input image, for example converting an input image taken midday to a nice sunset image. They first retrieve the most similar video frame to input scene from dataset as reference frame. Then they find matching patches between reference frame and input image. Lastly, they transfer the variation occurs between reference frame and desired reference frame which is same scene but taken different time of day to input image. Laffont et al. [41] take a step forward in their work for handling more general variations as transient attributes such as lighting, weather, and seasons.

High-level image editing offers easier and more natural way to casual users to manipulate a given image. Instead of using a reference image either provided by the user or retrieved from a database, learning the image manipulations and high-level attributes for image editing like a human has also attracted researchers. Berthouzoz et al. [111] learn parameters of the basic
operations for some manipulations recorded in Photoshop as macro to adapt them to new images, for example, applying same skin color correction operation with same parameters for both faces with dark-skinned and light-skinned does not give expected correction. In contrast to learning image operations for specific editing effects, Cheng et al. [112] learn the attributes as adjectives and objects as nouns for semantic parsing of an image and further use them for verbal guided image manipulation to indoor images. For example, the verbal command “change the floor to wooden” modifies the appearance of the floor. Similarly, Laffont et al. [41] learn to recognize transient attributes for attribute-guided image editing on outdoor images. To modify the look of an input image (e.g. a photo taken in a sunny day), they first locate similar scenes in a dataset they collected and annotated with transient attributes. Then they transfer the desired look (e.g. “more winter”) from the corresponding version of the candidate match images by using an appearance transfer method. Lee et al. [113] aim to automatically select a subset of style exemplars that will achieve good stylization results by learning a content-to-style mapping between large photo collection and a small style dataset.

Deep learning has fueled a growing literature on employing neural approaches to improve existing image editing problems. Here, we review the studies that are the most relevant to our work. Gatys et al. [114] has demonstrated how Convolutional Neural Networks (CNNs) effectively encode content and texture separately in feature maps of CNNs trained on large-scale image datasets and has proposed a neural style transfer method to transfer artistic styles from paintings to natural images. Alternatively, Johnson et al. [104] train a transformation network to speed up the test time of style transferring together with minimization of perceptual loss between input image and stylized image. Li et al. [115] consider a deep feed-forward network, which is capable of generating multiple and diverse results within a single network. Recent deep photo style transfer method of Luan et al. [42], named DPST, aims at providing realism in case of style transfer is made between the real photos. For example, when one wants to make an input photo look like taken in different illumination and weather conditions, a photo-realistic transfer is necessary. It uses semantic labels to prevent semantic inconsistency so that style transfer is carried out between same semantic regions. Recently, Li et al. [43] have proposed another photo style transfer method
called FPST, which works significantly faster than DPST. It considers a two-steps process, a stylization step followed by a photorealistic smoothing step, both of each having efficient closed-form solutions. There are some style transfer networks which are specialized for the editing face images and portraits [116–118] with new objectives. Nevertheless, these style transfer works limit the users to find an reference photo in which desired style effects exist for desired attributes.

Yan et al. [119] introduce the first automatic photo adjustment framework based on deep neural networks. They use deep neural network to learn a regressor which transforms the colors for artistic styles especially color adjustment from the image and its stylized version pairs. They define a set of feature descriptors based on pixel, global and semantic levels. In another work, Gharbi et al. [120] propose a new neural network architecture to learn image enhancement transformations at low resolution, then they move learned transformations to higher resolution in bilateral space in an edge-preserving manner.

Lastly, building upon conditional GAN model, some image completion works have been proposed to predict missing regions providing global and local context information with multiple discriminator networks [121, 122].

3.3. **ALS18K Dataset**

To train our model, we curate a new dataset by selecting and annotating images from two popular scene datasets, namely ADE20K [55] and Transient Attributes [41], for the reasons which will become clear shortly.

ADE20K [55] includes 22,210 images from a diverse set of indoor and outdoor scenes which are densely annotated with object and stuff instances from 150 classes. However, it does not include any information about transient attributes. Transient Attributes [41] contains 8,571 outdoor scene images captured by 101 webcams in which the images of the same scene can exhibit high variance in appearance due to variations in atmospheric conditions caused by weather, time of day, season. The images in this dataset are annotated with 40 transient scene
attributes, e.g. sunrise/sunset, cloudy, foggy, autumn, winter, but this time it lacks semantic layout labels.

To establish a richly annotated, large-scale dataset of outdoor images with both transient attribute and layout labels, we further operate on these two datasets as follows. First, from ADE20K, we manually pick the 9,201 images corresponding to outdoor scenes, which contain nature and urban scenery pictures. For these images, we need to obtain transient attribute annotations. To do so, we conduct initial attribute predictions using the pretrained model from [123] and then manually verify the predictions. From Transient Attributes, we select all the 8,571 images. To get the layouts, we first run the semantic segmentation model by Zhao et al. [124], the winner of the MIT Scene Parsing Challenge 2016, and assuming that each webcam image of the same scene has the same semantic layout, we manually select the best semantic layout prediction for each scene and use those predictions as the ground truth layout for the related images.

In total, we collect 17,772 outdoor images (9,201 from ADE20K + 8,571 from Transient Attributes), with 150 semantic categories and 40 transient attributes. Following the train-val split from ADE20K, 8,363 out of the 9,201 images are assigned to the training set, the other 838 testing; for the Transient Attributes dataset, 500 randomly selected images are held out for testing. In total, we have 16,434 training examples and 1,338 testing images. Lastly, we resize the height of all images to 512 pixels and apply center-cropping to obtain $512 \times 512$ images. In the following section, we discuss our data collection efforts for our ALS18K dataset in more detail.

### 3.3.1. Dataset Collection

Fig. 3.2. presents example semantic layout predictions for some images from the Transient Attribute dataset [41], obtained with the method in [124]. In a similar fashion, Fig. 3.3. illustrates transient attributes estimated by the network in [123] for some images from the ADE20K dataset [55]. Finally, Fig. 3.4. shows the distribution of the most frequent object classes in our proposed ALS18K dataset, sorted by their number of occurrences.
3.4 Attribute Manipulation Framework

Our framework provides an easy and high-level editing system to manipulate transient attributes of outdoor scenes (see Fig. 3.5.). The key component of our framework is a scene generation network that is conditioned on semantic layout and continuous-valued vector of transient attributes. This network allows us to generate synthetic scenes consistent with the semantic layout of the input image and having the desired transient attributes. One can play with 40 different transient attributes by increasing or decreasing values of certain dimensions. Note that, at this stage, the semantic layout of the input image should also be fed to the network, which can be easily automated by a scene parsing model. Once an artificial scene with desired properties is generated, we then transfer the look of the hallucinated image to the original input image to achieve attribute manipulation in a photorealistic manner.

In Section 3.4.1., we present the architectural details of our attribute and layout conditioned scene generation network and the methodologies for effectively training our network. Finally, in Section 3.4.2., we discuss the photo style transfer method that we utilize to transfer the appearance of generated images to the input image.
3.4.1. Scene Generation

In this section, we first give a brief technical summary of GANs and conditional GANs (CGANs), which provides the foundation for our scene generation network (SGN). We then present architectural details of our SGN model, followed by the two strategies applied for improving the training process. All the implementation details are included in the Supplementary Material.
<table>
<thead>
<tr>
<th>Object</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>sky</td>
<td>12188</td>
</tr>
<tr>
<td>tree</td>
<td>12188</td>
</tr>
<tr>
<td>building</td>
<td>12188</td>
</tr>
<tr>
<td>mountain</td>
<td>7929</td>
</tr>
<tr>
<td>road</td>
<td>7929</td>
</tr>
<tr>
<td>grass</td>
<td>7929</td>
</tr>
<tr>
<td>car</td>
<td>7929</td>
</tr>
<tr>
<td>person</td>
<td>7929</td>
</tr>
<tr>
<td>wall</td>
<td>7929</td>
</tr>
<tr>
<td>sidewalk</td>
<td>7929</td>
</tr>
<tr>
<td>fence</td>
<td>7929</td>
</tr>
<tr>
<td>plant</td>
<td>7929</td>
</tr>
<tr>
<td>signboard</td>
<td>7929</td>
</tr>
<tr>
<td>voter</td>
<td>7929</td>
</tr>
<tr>
<td>sea</td>
<td>7929</td>
</tr>
<tr>
<td>streetlight</td>
<td>7929</td>
</tr>
<tr>
<td>yard</td>
<td>1686</td>
</tr>
<tr>
<td>pole</td>
<td>1686</td>
</tr>
<tr>
<td>sand</td>
<td>1686</td>
</tr>
<tr>
<td>rock</td>
<td>1686</td>
</tr>
<tr>
<td>land</td>
<td>1686</td>
</tr>
<tr>
<td>path</td>
<td>1686</td>
</tr>
<tr>
<td>hill</td>
<td>1686</td>
</tr>
<tr>
<td>floor</td>
<td>1686</td>
</tr>
<tr>
<td>house</td>
<td>1686</td>
</tr>
<tr>
<td>river</td>
<td>1686</td>
</tr>
<tr>
<td>truck</td>
<td>1686</td>
</tr>
<tr>
<td>bridge</td>
<td>1686</td>
</tr>
<tr>
<td>bush</td>
<td>1686</td>
</tr>
<tr>
<td>van</td>
<td>1686</td>
</tr>
<tr>
<td>runway</td>
<td>1686</td>
</tr>
<tr>
<td>stove</td>
<td>1686</td>
</tr>
<tr>
<td>railing</td>
<td>1686</td>
</tr>
<tr>
<td>door</td>
<td>1686</td>
</tr>
<tr>
<td>bench</td>
<td>1686</td>
</tr>
<tr>
<td>awning</td>
<td>1686</td>
</tr>
<tr>
<td>ashcan</td>
<td>1686</td>
</tr>
</tbody>
</table>

Figure 3.4. Distribution of the most frequent object classes in our proposed ALS18K dataset (sorted by their number of occurrences).

**Background.**

In Generative Adversarial Networks (GANs) [25], a discriminator network \( D \) and a generator network \( G \) play a two-player min-max game where \( D \) learns to determine if an image is real or fake and \( G \) strives to output as realistic images as possible to fool the discriminator. The \( G \) and \( D \) are trained jointly by performing alternative updates:

\[
\min_G \max_D \mathcal{L}_{GAN}(G, D) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log (1 - D(G(z)))]
\]

where \( x \) is a natural image drawn from the true data distribution \( p_{data}(x) \) and \( z \) is a random noise vector sampled from a multivariate Gaussian distribution. The optimal solution to this
Figure 3.5. Overview of the proposed attribute manipulation framework. Given an input image and its semantic layout, we first resize and center-crop the layout to $512 \times 512$ pixels and feed it to our scene generation network. After obtaining the scene synthesized according to the target transient attributes, we transfer the look of the hallucinated style back to the original input image.

The min-max game is when the distribution $p_G$ converges to $p_{\text{data}}$.

Conditional GANs [37] (CGANs) engage additional forms of side information as generation constraints, e.g. class labels [37], image captions [28], bounding boxes and object keypoints [29]. Given a context vector $c$ as side information, the generator $G(z, c)$, taking both the random noise and the side information, tries to synthesize a realistic image that satisfies the condition $c$. The discriminator, now having real/fake images and context vectors as inputs, aims at not only distinguishing real and fake images but also whether an image satisfies the paired condition $c$. Such characteristics is referred to as match-aware [28]. In this way, we expect the generated output of CGAN $x_g$ is controlled by the side information $c$. Particularly, in our model, $c$ is composed of semantic layouts $s$ and transient attributes $a$.

Proposed Architecture

In our work, we follow a multi-scale strategy similar to that in Pix2pixHD [40]. Our scene generator network (SGN), however, takes the transient scene attributes and a noise vector as extra inputs in addition to the semantic layout. While the noise vector provides stochasticity and controls diversity in the generated images, transient attributes let the users have control on the generation process. In more detail, our multi-scale generator network $G = \{G_1, G_2\}$
consists of a coarse-scale ($G_1$) generator and a fine-scale ($G_2$) generator. As illustrated in Fig. 3.6., $G_1$ and $G_2$ have nearly the same architecture, with the exception that they work on different image resolutions. While $G_1$ operates at a resolution of $256 \times 256$ pixels, $G_2$ outputs an image with a resolution that is $4 \times$ larger, i.e. $512 \times 512$ pixels. Here, the image generated by $G_1$ is fed to $G_2$ as an additional input in the form of a tensor. In that regard, $G_2$ can be interpreted as a network that performs local enhancements in the fine resolution.

In our coarse and fine generator networks, while the semantic layout categories are encoded into 8-bit binary codes, transient attributes are represented by a 40-d vector. Input semantic layout map $S$ is of the same resolution with our fine scale image resolution. We concatenate semantic layout $S$ and noise $z$, and feed their concatenation into convolutional layers of $G_1$ and $G_2$ to obtain semantic feature tensors, which are used as input to the subsequent residual blocks. For the coarse scale generator $G_1$, we at first perform a downsampling operation with a factor of 2 to align the resolutions. Then, spatially replicated attribute vectors $a$ are concatenated to input tensors of each residual block in $G_1$ and $G_2$ to condition the image generation process in regard to input transient scene attributes. Finally, deconvolutional layers are used to upsample the feature tensor of the last residual block to obtain final output images. For fine scale generator $G_2$, semantic feature tensor extracted with the convolutional layers is summed with the feature tensor from the last residual block of coarse generator $G_1$ before feeding into residual blocks of fine scale generator $G_2$.

The discriminator used in our SGN also adopts a multi-scale approach in that it includes three different discriminators denoted by $D_1, D_2, D_3$ with similar network structures that operate at different image scales. In particular, we create an image pyramid of 3 scales that include real and generated high resolution images, their downsampled versions by a factor of 2 and 4. Our discriminators take tuples of real or synthesized images from different levels of this image pyramid, matching or mismatching semantic layouts and transient attributes and
Figure 3.6. Scene Generation Network (SGN). Our proposed CGAN architecture for generating synthetic outdoor scenes consistent with given layout and transient attributes.

decide whether the images are fake or real, and whether the pairings are valid. That is,

\[ D_k(x_k, a, S) = \begin{cases} 
1, & x_k \in p_{\text{data}} \text{ and } x_k, a, S \text{ correctly match,} \\
0, & \text{otherwise.} 
\end{cases} \]

with \( k = \{1, 2, 3\} \) denoting image scales. Hence, the training our conditional GAN models becomes a multi-task learning problem defined as follows:

\[
\min_G \max_{D_1, D_2, D_3} \sum_{k=\{1,2,3\}} L_{GAN}(G, D_k) \tag{24}
\]

The architectural details of our Scene Generation Network are given in Table 3.1. In this table, we follow a naming convention similar to the one used in [33, 40]. For instance, \( C_{3}128S_{2} \) denotes a Convolution-InstanceNorm-ReLU layer with 128 filters of kernel size \( 3 \times 3 \) kernel and stride 2. \( f_{1i} \) and \( f_{2i} \) represent \( i \)th internal feature tensors of \( G_1 \) and \( G_2 \), respectively. \( R512 \) denotes a residual block with filter size 512 as depicted in Fig. 3.6. Similarly, \( D_{3}128S_{0.5} \) represents a Deconvolution-InstanceNorm-ReLU layer with 128 filters of kernel size \( 3 \times 3 \) and stride 0.5. At the last deconvolution layer \( D_{7}3S_{1} \), we do not use InstanceNorm and replace ReLU activations with \( \tanh \). The discriminator resembles a Siamese network [125, 126], where one stream takes the real/generated image as input \( x \) and
Table 3.1. Architectural details of the generator and discriminator networks.

<table>
<thead>
<tr>
<th>Generator</th>
<th>Input Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z \parallel S$</td>
<td>$C_7 64S_1 - C_2 128S_2 - C_2 256S_2 - C_2 512S_2 \rightarrow f_{11}$</td>
</tr>
<tr>
<td>$f_{11} \parallel a$</td>
<td>$R512 - R512 - R512 - R512 - R512 \rightarrow f_{12}$</td>
</tr>
<tr>
<td>$f_{12}$</td>
<td>$D_3 256S_{0.5} - D_3 128S_{0.5} - D_3 64S_{0.5} \rightarrow f_{13}$</td>
</tr>
<tr>
<td>$f_{13}$</td>
<td>$D_7 3S_1 \rightarrow x_{\text{fake}}$</td>
</tr>
</tbody>
</table>

$G_2$  $f_{13} + f_{21}$  $R64 - R64 \rightarrow f_{22}$  $f_{22}$  $D_3 64S_{0.5} - D_7 3S_1 \rightarrow x_{\text{fake}}$

<table>
<thead>
<tr>
<th>Discriminator</th>
<th>Input Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>$C_4 64S_2 - C_4 128S_2 - C_4 256S_2 - C_4 512S_2 \rightarrow f_x$</td>
</tr>
<tr>
<td>$a \parallel S$</td>
<td>$C_4 64S_2 - C_4 128S_2 - C_4 256S_2 - C_4 512S_2 \rightarrow f_c$</td>
</tr>
<tr>
<td>$f_x \parallel f_c$</td>
<td>$C_1 512S_1 - C_4 1S_1 \rightarrow {0, 1}$</td>
</tr>
</tbody>
</table>

the second one processes the given attributes $a$ and the spatial layout labels $S$. The responses of these networks are then concatenated $a \parallel S$ and fused via a $1 \times 1$ convolution operation. The combined features are finally sent to fully-connected layers for the binary decision. We use leaky ReLU with slope 0.2 for our discriminator networks. We do not use InstanceNorm at the input layers. We employ 3 discriminators at 3 different spatial scales with 1, 0.5 and 0.25 as the scaling factors for both coarse and fine scale generators $G_1$ and $G_2$ during training.

**Improved Training of SGNs**

Here we elaborate on two complementary training techniques that substantially boost the efficiency of the training process.

**Relevant Negative Mining.** Training the match-aware discriminator in CGAN resembles learning to rank [127], in the sense that a “real pair”–real image paired with right conditions–should score higher (i.e. classifying into category 1 in this case) than a “fake pair”–either image is fake or context information is mismatched (i.e. classifying into category 0). For ranking loss, it has been long acknowledged that naively sampling random
negative examples is inferior to more carefully designed negative sampling scheme, such as various versions of hard negative mining [100–103]. Analogously, a better negative mining scheme can be employed by training CGAN, as existing works have been using random sampling [29]. To this end, we propose to apply the concept of relevant negative mining [102] (RNM) to sample mismatching layout in training our SGN model. Concretely, for each layout $S$, we search for its nearest neighbor $S'$ and set it as the corresponding mismatching negative example for $S$. In Section 3.5., we present empirical qualitative and quantitative results to demonstrate improvement from RNM over random sampling. We attempted similar augmentation on attributes $a$ by flipping a few of them instead of complete random sampling to obtain the mismatching $a'$ but found such operation hurt the performance, likely due to the flipped attributes being too semantically close to the original ones which cause ambiguity to the discriminator.

**Layout-Invariant Perceptual Loss.** Following the practice of existing works [34, 89], we also seek to stabilize adversarial training and enhance generation quality by adding a perceptual loss. Conventionally, features used for perceptual loss come from a deep CNN, such as VGG [128], pretrained on ImageNet for classification task. However, perceptual loss to match such features would intuitively withhold generation diversity, which opposes our intention of creating stochastic output via a GAN framework. Instead, we propose to employ intermediate features trained on outdoor scene parsing with ADE20K. The reason for doing so is three-fold: diversity in generation is not suppressed, because scenes with different contents but the same layout ideally produce the same high-level features; the layout of the generation is further enforced thanks to the nature of the scene parsing network; since the scene parsing network is trained on real images, the perceptual loss will impose additional regularization to make the output more photorealistic. The final version of our proposed perceptual loss is as follows:

$$
\mathcal{L}_{\text{percep}}(G) = E_{z \sim p(z) ; x, S, a \sim p_{\text{data}}(S, a)} \left[ \| f_P(x) - f_P(G(z, a, S)) \|^2_2 \right],
$$

(25)
where $f$ is the CNN encoder for the scene parser network. Our full objective that combines multi-scale GAN loss and layout-invariant feature matching loss thus becomes:

$$\min_G \left( \max_{D=(D_1,D_2,D_3)} \sum_{k=1,2,3} \mathcal{L}_{GAN}(G, D_k) + \lambda \mathcal{L}_{percep}(G) \right) \tag{26}$$

where $\lambda$ is a scalar controlling the importance of our proposed layout-invariant feature matching loss and is set to 10 in our experiments. By additionally considering RNM and perceptual loss, we arrive at the training procedure which is outlined in Algorithm 1.

### 3.4.2. Style Transfer

The main goal in photo style transfer is to successfully transfer visual style (such as color and texture) of a reference image onto another image while preserving semantic structure of the target image. In the past, statistical color transfer methods [105, 106] showed that the success of the style transfer methods highly depend on the semantic similarity of the source and target images. To overcome this obstacle, user interaction, semantic segmentation

---

**Algorithm 1 SGN training algorithm**

1. **Input:** Training set $\Omega = \{(x, a, S)\}$ with training images $x$, semantic segmentation layouts $S$ and transient attributes $a$.
2. **for all** number of iterations **do**
3. sample minibatch of paired $x, a, S$
4. sample minibatch of $z_i$ from $\mathcal{N}(0, I)^d$
5. **for all** $(x_i, a_i, S_i)$ in $\Omega$ **do**
6. Randomly sample negative $a'_i$ mismatching $x_i$
7. Sample $S'_i$ mismatching $x_i$ via RNM
8. **end for**
9. $x_g \leftarrow G(z_i, a_i, S_i)$ \{Forward through generator\}
10. **for** $k=1:3$ **do**
11. $\mathcal{L}_{D_k} \leftarrow -(\log D_k(x, a, S) + \log (1 - D_k(x, a, S)) + \log (1 - D_k(x, a', S'))$
12. $D_k \leftarrow D_k - \alpha \partial \mathcal{L}_{D_k} / \partial D_k$ \{Update discriminator $D_k$\}
13. **end for**
14. $\mathcal{L}_G \leftarrow - \log D(x, a, S) + \lambda \| f_p(x) - f_p(x_g) \|_2^2$
15. $G \leftarrow G - \alpha \partial \mathcal{L}_G / \partial G$ \{Update generator $G$\}
16. **end for**
approaches or image matching methods were utilized to provide semantic relation between
source and target images. In addition, researchers explored data driven methods to come
up with fully automatic approaches which retrieve the source style image through some
additional information such as attributes, features and semantic similarity.

For existing deep learning based photo style transfer methods, it is still crucial that source
and reference images have similar semantic layouts to provide successful and realistic
style transfer results. Image retrieval based approaches are limited with the dataset and
they become infeasible when there is no images with the desired properties. The key
distinguishing characteristics of our framework is that we can generate a style image on
the fly that has both similar semantic layout with the input image and possess the desired
transient attributes, thanks to our proposed SGN model. In our framework, for photo style
transfer, we consider employing both DPST [42] and FPST [43] models.

DPST [42] extends the formalization of the neural style transfer method of Gatys et al. [114]
by adding a photorealism regularization term that enables the style transfer to be done
between same semantic regions instead of the whole image. This property makes DPST very
appropriate for our image manipulation system. Although this method in general produces
fairly good results, we observe that it sometimes introduces some smoothing and visual
artifacts in the output images, which hurt the photorealism. For that reason, we first apply
a cross bilateral filter [129] to smooth the DPST’s output according to edges in the input
image and then apply the post-processing method proposed by Mechrez et al. [130], which
uses screened Poisson equation to make the stylized image more similar to the input image
in order to increase its visual quality.

FPST [43] formulates photo style transfer as a two steps procedure. The first step carries
out photorealistic image stylization by using a novel network architecture motivated by the
whitening and coloring transform [131], in which the upsampling layers are replaced with
unpooling layers. The second step performs a manifold ranking based smoothing operation
to eliminate the structural artifacts introduced by the first step. As both of these steps have
closed-form solutions, FPST works much faster than DPST. Since FPST involves an inherent
smoothing step, in our experiments, we only apply the approach by Mechrez et al. [130] as a post-processing step.

3.5. Results and Comparison

We first evaluate our scene generation network’s ability to synthesize diverse and realistic-looking outdoor scenes, then show attribute manipulation results of our proposed two-stage framework that employs the hallucinated scenes as reference style images. Lastly, we discuss the limitations of the approach.

3.5.1. Attribute and Layout Guided Scene Generation

Here, we assess the effectiveness of our SGN model on generating outdoor scenes in terms of image quality, condition correctness and diversity. We also demonstrate how the proposed model enables the users to add and subtract scene elements.

Training Details

All models were trained with a mini-batch size of 40 where parameters were initialized from a zero-centered Gaussian distribution with standard deviation of 0.02. We set the amount of the layout-invariant feature matching loss $\lambda$ to 10. We used the Adam optimizer [132] with the learning rate value of $2 \times 10^{-4}$ and the momentum value of 0.5. For data augmentation, we employed horizontal flipping with a probability of 0.5. We trained our coarse-scale networks for 100 epochs on a NVIDIA Tesla K80 GPU for 3 days. After training them, we kept their parameters fixed and trained our fine-scale networks for 10 epochs. Then, in the next 70 epochs, we updated the parameters of both of our fine and coarse-scale networks together. Our implementation is based on the PyTorch framework. Training of our fine-scale networks took about 10 days on a single GPU.
Ablation Study

We illustrate the role of Relevant Negative Mining (RNM) and layout-invariant Perceptual Loss (PL) in improving generation quality with an ablation study. Here we consider the outputs of the coarse-scale generator $G_1$ to evaluate these improvements as it acts like a global image generator. Our input layouts come from the test set, i.e. are unseen during training. Furthermore, we fix the transient attributes to the predictions of the pre-trained deep transient model [123]. Fig. 3.7. presents synthetic outdoor images generated from layouts depicting different scene categories such as urban, mountain, forest, coast, lake and highway. We make the following observations from these results. Attributes of the generated images are mostly in agreement with the original transient attributes. Integrating RNM slightly improves the rendering of attributes but in fact, its main role is to make training more stable. Our proposed layout-invariant PL boosts the final image quality of SGN. The roads, the trees and the clouds are drawn with the right texture; the color distributions of the sky, the water and the field also appear realistic; reasonable physical effects are also observed such as the reflection of the water, fading of the horizon, valid view perspective of urban objects. In our analysis, we also experimented with the VGG-based perceptual loss, commonly employed in many generative models, but as can be seen from Fig. 3.7., our proposed perceptual loss, which performs feature matching over a pretrained segmentation network, gives much better results in terms of photorealism. Overall, the results with both RNM and PL are visually more pleasing and faithful to the attributes and layouts.

For quantitative evaluation, we employ the Inception Score (IS) [92] and the Fréchet Inception Distance (FID) [133]\(^5\)

The IS correlates well with human judgment of image quality where higher IS indicates better quality. FID has been demonstrated to be more reliable than IS in terms of assessing the realism and variation of the generated samples. Lower FID value means that the distributions of generated images and real images are similar to each other. Table 3.2. shows the IS and

\(^5\)In our evaluation, we utilized the official implementations of IS and FID. IS scores are estimated by considering all of the test images from our dataset, which were not seen during training and by using a split size of 10. While calculating FID scores, we employ all of the test images from our dataset as the reference images.
Figure 3.7. Sample scene generation results. In these examples, the input layouts are from the test set, which are unseen during training and the transient attributes are fixed to the original transient attributes. Incorporating Relevant Negative Mining (RNM) and Perceptual Loss (PL) significantly improves the performance of the baseline SGN model in terms of both image quality as well as faithfulness of the end result to conditioned layouts and attributes. Moreover, the way we define our perceptual loss, as compared to commonly used VGG-based one, provides better and more photorealistic results.

FID values for our SGN model trained under various settings, together with values for the real image space. These results agree with our qualitative analysis that training with RNM and Perceptual Loss provides samples of the highest quality. Additionally, for each generated image, we also predict its attributes and semantic segmentation map using separately trained attribute predictor by Baltenberger et al. [123] and the semantic segmentation model by Zhou et al. [55] and we report the average $\text{MSE}^6$ and segmentation accuracy again in Table 3.2.

Training with the proposed perceptual loss is more effective in reflecting photorealism and preserving both the desired attributes and the semantic layout better than the VGG-based perceptual loss.

Our SGN model with RNM and Perceptual Loss shows clear superiority to other variants both qualitatively and quantitatively. Thus from now on, if not mentioned otherwise, all of our results are obtained with this model.

---

6The ground truth attributes are scalar values between 0 and 1.
Table 3.2. Ablation study. We compare visual quality with respect to Inception Score (IS) and Fréchet Inception distance (FID), attribute and semantic layout correctness in terms of average MSE of attribute predictions (Att. MSE) and segmentation accuracy (Seg. Acc.), respectively, via pre-trained models. Our SGN model trained with RNM and PL techniques consistently outperforms the others, including the setting with VGG-based perceptual loss.

<table>
<thead>
<tr>
<th>Model</th>
<th>IS</th>
<th>FID</th>
<th>Att. MSE</th>
<th>Seg. Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGN</td>
<td>3.91</td>
<td>43.77</td>
<td>0.016</td>
<td>67.70</td>
</tr>
<tr>
<td>+RNM</td>
<td>3.89</td>
<td>41.84</td>
<td>0.016</td>
<td>70.11</td>
</tr>
<tr>
<td>+VGG</td>
<td>3.80</td>
<td>41.87</td>
<td>0.016</td>
<td>67.42</td>
</tr>
<tr>
<td>+PL</td>
<td>4.15</td>
<td>36.42</td>
<td>0.015</td>
<td>70.44</td>
</tr>
<tr>
<td>+RNM+PL</td>
<td>4.19</td>
<td>35.02</td>
<td>0.015</td>
<td>71.80</td>
</tr>
<tr>
<td>Original</td>
<td>5.77</td>
<td>0.00</td>
<td>0.010</td>
<td>75.64</td>
</tr>
</tbody>
</table>

Comparison with Image-to-Image Translation Models

We compare our model to Pix2pix [39] and Pix2pixHD [40] models. It is worth mentioning that both of these two approaches generate images only by conditioning on the semantic layout but not transient attributes, and moreover, they do not utilize noise vectors. We provide qualitative comparisons in Fig. 3.8. As these results demonstrate, our model not only generates realistic looking images on par with Pix2pixHD but also has the capability to deliver control over the attributes of the generated scenes. “Sunset” attribute makes the horizon slightly more reddish, “Dry” attribute increases the brown tones on the trees, “Snow” attribute whitens the ground. Also note that the emergence of each attribute tends to highly resonate with part of the image that is most related to the attribute. That is, “Clouds” attribute primarily influences the sky, whereas “Winter” attribute correlates with the ground, and “Lush” tends to impact the trees and the grass. This further highlights our model’s reasoning capability about the attributes in producing realistic synthetic scenes. In Figure 3.9., we present additional comparisons of our SGN model to Pix2pixHD [40] method.

For quantitative comparison, we compare the IS and FID scores and segmentation accuracy using all 1,338 testing images in Table 3.3. considering both coarse and fine scales. These results suggest that our proposed model produces high fidelity natural images better than Pix2pixHD in both scales. The difference in the segmentation accuracy suggests that

---

7For both of these models, we use the original source codes provided by the authors.
Figure 3.8. Comparison of our SGN model against Pix2pix [39] and Pix2pixHD [40]. Each row shows the original image and the samples generated according to its corresponding semantic layout. Since our SGN model also takes into account a set of target transient attributes (only the top three most significant ones are shown here for the sake of simplicity), it can generate diverse and more realistic results than the other methods.

Pix2pixHD puts a more strict restraint on the layout whereas our model offers flexibility in achieving a reasonable trade-off between capturing realism in accordance with transient attributes vs. fully agreeing with the layout. Furthermore, in addition to these metrics, we conduct a human evaluation on Figure Eight\textsuperscript{8}, asking workers to select among the results of our proposed model and the Pix2pixHD method (for the same semantic layout) which they believe is more realistic. We randomly generate 200 questions, and let 5 different subjects answer each question. We did not set a time limit to the workers to make their decisions. A screenshot of our user interface is shown in Fig. 3.10. Moreover, Fig. 3.11. outlines the demographics of our participants. The majority of them were between the ages of 35 and 44 years old, the youngest being 18 and the oldest being 65. The gender ratio was skewed towards males (67% males and 33% females), and most of them have no technical expertise (28% had no specific interest, 61% were hobbyist, 11% were working on image processing/computer graphics.

We find that 66% of the subjects picked our results as more realistic. These results suggest

\textsuperscript{8}Figure Eight is a web-based data annotation company which can be accessed from \url{https://www.figure-eight.com/}

61
Figure 3.9. Comparison of the proposed SGN against Pix2pixHD. Given the input semantic layout, we show the reference image this semantic layout belongs to and the synthetic images generated by Pix2pixHD and our SGN model.
that besides the advantages of manipulation over transient attributes, our model also produces higher quality images than the Pix2pixHD model. We also compared our results to the recently proposed Cascaded Refinement Network [34], however, it did not give meaningful results on our dataset with complex scenes\footnote{We trained this model using the official code provided by the authors.}.

Figure 3.10. A screenshot of a sample question in our user study on scene synthesis.
Figure 3.11. (a) Age, (b) gender and (c) profession distribution of the participants of our user study on scene synthesis.

Table 3.3. Quantitative comparison of layout conditioned image synthesis approaches. Our model consistently outperforms others in both coarse and fine resolutions in terms of photorealism, as measured by IS and FID.

<table>
<thead>
<tr>
<th>Model</th>
<th>IS</th>
<th>FID</th>
<th>Seg. Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coarse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pix2pix</td>
<td>3.26</td>
<td>76.40</td>
<td>61.93</td>
</tr>
<tr>
<td>Pix2pixHD</td>
<td>4.20</td>
<td>47.86</td>
<td>75.57</td>
</tr>
<tr>
<td>Ours</td>
<td><strong>4.19</strong></td>
<td><strong>35.02</strong></td>
<td>71.80</td>
</tr>
<tr>
<td>Original</td>
<td>5.77</td>
<td>0.00</td>
<td>75.64</td>
</tr>
<tr>
<td>Fine</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pix2pixHD</td>
<td>4.87</td>
<td>50.85</td>
<td><strong>76.17</strong></td>
</tr>
<tr>
<td>Ours</td>
<td><strong>5.05</strong></td>
<td><strong>36.34</strong></td>
<td>74.60</td>
</tr>
<tr>
<td>Original</td>
<td>7.37</td>
<td>0.00</td>
<td>77.14</td>
</tr>
</tbody>
</table>

Diversity of the Generated Images

In our framework, a user can control the diversity via three different mechanisms, each playing a different role in the generation process. Perhaps the most important one is the input semantic layout which explicitly specifies the content of the synthesized image, and the other two are the target transient attributes and the noise vector. In Fig. 3.12., we show the effect of varying the transient attributes for a sample semantic layout and Fig. 3.13. illustrates the role of noise. If we keep the layout and the attributes fixed, the random noise vector mainly affects the appearance of some local regions, especially the ones involving irregular or stochastic textures such as the sky, the trees or the plain grass. The transient attribute vectors, however, has a more global effect, modifying the image without making any changes to the constituent parts of the scene.
Figure 3.12. Modifying transient attributes in generating outdoor images under different weather and time conditions. Our model’s ability of varying with transient attributes contributes to the diversity and photorealism in its generation (more results can be found in the Supplementary Material).

A synthetic scene  Stochastic variations  Standard deviation

Figure 3.13. The effect of varying the noise vector. For an example synthetically generated scene (left), we show close-up views from three different regions (middle) from samples obtained with only changing the random noise. Standard deviation of each pixel over 100 different realizations of the scene (right), which demonstrates that the random noise causes stochastic variations within the irregular or stochastic textural regions.

Adding and Subtracting Scene Elements

Here we envision a potential application of our model as a scene editing tool that can add or subtract scene elements. Fig. 3.14. demonstrates an example. We begin with a coarse spatial layout which contains two large segments denoting the “sky” and the “ground”. We then gradually add new elements, namely “mountain”, “tree”, “water”. At each step, our
Figure 3.14. Gradually adding and removing elements to and from the generated images. We use a coarse spatial layout map (top left) to generate an image from scratch, and then keep adding new scene elements to the map to refine the synthesized images. Moreover, we also show how we can modify the look by conditioning on different transient attributes.

3.5.2. Attribute Transfer

We demonstrate our attribute manipulation results in Fig 3.15.. Here we provide results obtained by using FPST [43] as it gives slightly better results in our experiments and also significantly faster than DPST [42]. We also provide the results of both FPST and DPST in the following sections. As can be seen, our algorithm produces photorealistic manipulation results for many different types of attributes like “Sunset”, “Spring”, “Fog”, “Snow”, and moreover, a distinctive property of our approach is that it can perform multimodal editing for a combination of transient attributes as well, such as “Winter and Clouds” and “Summer and Moist”. It should be noted that modifying an attribute is inherently coupled with the appearance of certain semantic scene elements. For example, increasing “Winter” attribute makes the color of the grass white whereas increasing “Autumn” attribute turns them to brown. As another example, “Clouds” attribute does not modify the global appearance of the
Figure 3.15. Sample attribute manipulation results. Given an outdoor scene and its semantic layout, our model produces realistic looking results for modifying various different transient attributes. Moreover, it can perform multimodal editing as well, in which we modify a combination of attributes.

scene but merely the sky region, comparing with “Fog” attribute which blurs distant objects; “Dry” attribute emphasizes the hot colors, while “Warm” attribute has the opposite effect. Some attributes such as “Fog”, however, have an influence on the global appearance.

In Figure 3.16. and Figure 3.17., we compare the performance of our method to the data-driven approach of Laffont et al. [41] using FPST and DPST respectively. As mentioned in Section 2, this approach first identifies a scene that is semantically similar to the input image using a database of images with attribute annotations, then it retrieves the version of that scene having the desired properties, and finally, the retrieved image is used as a reference for style transfer. For retrieving the images semantically similar to the source image we also use the Transient Attributes dataset and the retrieval strategy employed by Laffont et al. [41]. In fact, since the authors did not publicly share their attribute transfer code, in our experiments,
we consider the test cases provided in their project website. In the figure, we both present the reference images generated by our approach and retrieved by the competing method at the right-bottom corner of each output image.

Figure 3.16. Comparison with [41]. In each row, for a given input image (first column), we respectively provide the results of [41] using their exemplar-based style transfer method (second column) and FPST [43] (third column) between retrieved images and input images, and the results of our method (last column) using FPST [43] between generated image by proposed SGN model and input image.

For a fair comparison, we also present alternative results of [41] where we replace the original exemplar-based transfer method with FPST [43] or DPST [42], which is used in

---

10The test cases we used in our experimental analysis are available at http://transattr.cs.brown.edu/comparisonAppearanceTransfer/testCases.html.
obtaining our results\footnote{Note that, the post-processing method Mechrez et al. [130] is also employed here to improve photorealism.}. As can be seen, our approach produces better results than \cite{41} in terms of visual quality and as to reflecting the desired transient attributes. These results also demonstrate how style transfer methods are dependent on semantic similarity between the input and style images. Our main advantage over the approach by Laffont et al. \cite{41} is that the target image is directly hallucinated from the source image via the proposed SGN model, instead of retrieving the target image from a training set. This makes a difference...
since the source and the target images always share the same semantic layout. In this regard, our approach provides a more natural way to edit an input image to modify its look under different conditions.

Additionally, we conducted a user study on Figure Eight to validate our observations. We show the participants an input image and a pair of manipulation results along with a target attribute and force them to select one of the manipulated images which they consider visually more appealing regarding the specified target attribute. The manipulation results are either our results obtained by using DPST or FPST, or those of [41]. We have a total of 60 questions and we collected at least 3 user responses per each of these question. The manipulated images were the results of our framework obtained with DPST or FPST, or those obtained by the approach by Laffont et al. [41]. The users have unlimited time to make a selection. A screenshot of our user interface is shown in Fig. 3.18.

Fig. 3.19. summarizes the demographic distribution of the participants. The majority of them were between the ages of 25 and 34 years old, the youngest being 18 and the oldest being 59. The gender ratio was skewed towards males (66% males and 34% females), and most of them have no technical expertise (34% had no specific interest, 61% were hobbyist, 5% were working on image processing/computer graphics.. Table 3.4. summarizes these evaluation results. We find that the human subjects prefer our approach against the data-driven approach by [41] 65% of the time. This margin substantially increases when we replace the original exemplar-based transfer part of [41] with FPST as the semantic layouts of retrieved images are most of the time not consistent with those of the input images. We also evaluate the results of our frameworks with FPST and DPST being used as the style transfer network. As can be seen from Table 3.4., the human subjects prefers FPST against DPST but by a very small margin.

The most important advantage of our framework over existing works is that our approach enables users to play with the degree of desired attributes via changing the numerical values of the attribute condition vector. As shown in Figure 3.21., we can increase and decrease the strength of specific attributes and smoothly walk along the learned attribute manifold using
Table 3.4. User study results for attribute manipulation. The preference rate denotes the percentage of comparisons in which users favor one method over the other.

<table>
<thead>
<tr>
<th>Method Comparison</th>
<th>Preference rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours w/ FPST &gt; Laffont et al. [41]</td>
<td>65%</td>
</tr>
<tr>
<td>Ours w/ FPST &gt; Laffont et al. [41] w/ FPST</td>
<td>83%</td>
</tr>
<tr>
<td>Ours w/ FPST &gt; Ours w/ DPST</td>
<td>52%</td>
</tr>
</tbody>
</table>

Figure 3.18. A screenshot of a sample question in our user study on attribute transfer.

the outputs from the proposed SGN model. This is nearly impossible for a retrieval-based editing system since the style images are limited with the richness of the database.
Although our attribute manipulation approach is designed for natural images, we can apply it to oil paintings as well. In Figure 3.20, we manipulate transient attributes of three oil paintings to obtain their novel versions depicting these landscapes at different seasons. As can be seen from these results, our model also gives visually pleasing results for these paintings, hallucinating how they might look like if the painters picture the same scene at different times.
Figure 3.21. Our method can produce photorealistic manipulation results for different degrees of transient attributes. We give the results of both (a)FPST and (b)DPST photo style transfer methods.
**Effect of Post-Processing and Running Times**

We show the effects of the post-processing steps involved in our framework in Figure 3.22. As mentioned in Section 3.4.2., for DPST based stylized images, we first apply a cross bilateral filter (BLF) [129] and then employ screened Poisson equation (SPE) based photorealism enhancement approach [130]. For FPST based stylized images, we only apply SPE as it inherently performs smoothing. As can be seen from these results, the original stylized images demonstrate some texture artifacts and look more like a painting. Our post-processing steps make these stylized images photorealistic and more similar to the given input image.

In Table 3.5., we provide the total running time of our framework for manipulating the attributes of an outdoor image. There are three main parts, namely the scene generator network (SGN), the style transfer network, and the post-processing. We report the running time of each of these steps as well. For the style transfer and the post-processing steps, we employ two different versions, one depends on DPST and the other one depends on FPST.

![Figure 3.22. Effect of post-processing. Top: a sample input image and “Autumn” attribute transfer results by our framework with DPST [42] and FPST [43], respectively. Bottom: the impact of various post-processing strategies on the final results. See Section 3.4.2. for the details.](image)
and the corresponding smoothing operations. The experiment is conducted on a system with an NVIDIA Tesla K80 graphics card. We consider three different sizes for the input image and report the average run-time for each image resolution. Our FPST-based solution is, in general, much faster than our DPST-based one as most of the computation time is spent on the style transfer step. For images of $1024 \times 512$ pixels, while it takes 4 minutes to manipulate the attributes of an image with FPST, DPST requires 70 minutes to achieve the task.

Table 3.5. Running time analysis showing the average run time (in seconds) of each component of the proposed model across various image resolutions.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>SGN</th>
<th>Style Transfer</th>
<th>Post-Processing</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPST</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$512 \times 256$</td>
<td>0.10</td>
<td>1245.31</td>
<td>2.52</td>
<td>1247.93</td>
</tr>
<tr>
<td>$768 \times 384$</td>
<td>0.10</td>
<td>2619.48</td>
<td>4.61</td>
<td>2626.19</td>
</tr>
<tr>
<td>$1024 \times 512$</td>
<td>0.10</td>
<td>4130.27</td>
<td>7.24</td>
<td>4137.51</td>
</tr>
<tr>
<td>FPST</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$512 \times 256$</td>
<td>0.10</td>
<td>36.54</td>
<td>1.54</td>
<td>38.18</td>
</tr>
<tr>
<td>$768 \times 384$</td>
<td>0.10</td>
<td>99.34</td>
<td>3.63</td>
<td>103.07</td>
</tr>
<tr>
<td>$1024 \times 512$</td>
<td>0.10</td>
<td>222.20</td>
<td>6.22</td>
<td>228.52</td>
</tr>
</tbody>
</table>

3.5.3. Effect of Center-cropping

Our SGN works with a fixed resolution of $512 \times 512$ pixels and accepts the semantic layout of the center cropped and resized version of the input image. The style transfer networks consider SGN’s output as the target style image and manipulates the input image accordingly. When the image is very wide, like a panorama, center-cropping omits most of the source image. We analyze how this affects the overall performance our framework on a couple of panoramic images from SUN360 dataset [134]. We present attribute manipulation results for these images in Figure 3.23. We show additional results in Figure-3.24. We have observed that center-cropping does not pose a serious drawback to our approach, since the style transfer step exploits semantic layouts to constrain color transformations to be carried out between features from the image regions with the same label.
3.5.4. A Standalone GUI Prototype

We designed a GUI prototype that provides users with the ability to use our approach as an interactive photo editing tool. Fig. 3.25. presents a screenshot of our application. The GUI the layouts shows the input image and its semantic layout and it has controls to synthesize different versions of the scene based on desired set of transient attributes. Moreover, it allows users to control the degree of the attributes via the “increase” and “decrease” buttons. We also note that one can generate slightly different versions of the hallucinated scene by playing with the “Random Noise” button. Additionally, one can play with the layout to generate a novel scene from scratch.

3.5.5. Limitations

Our framework generally gives quite plausible results, but we should note that it might fail in some circumstances if either one of its components fails to function properly. In Fig. 3.26., we demonstrate such example failure cases. In the first row, the photo-realistic quality of the generated scene is not very high as it does not reproduce the houses well. As a consequence, the manipulation result is not very convincing. For the last two scenes, our SGN model
hallucinated “Fog” and “Night” attributes successfully but the style transfer network fails to transfer the looks to the input images.

Figure 3.24. Attribute manipulation results on panorama images.
Figure 3.25. A screenshot of the GUI prototype running our approach as a backend.

3.6. Discussion

We have presented a high-level image manipulation framework to edit transient attributes of natural outdoor scenes. The main novelty of the paper is to utilize a scene generation network in order to synthesize on the fly the reference style image that is consistent with the semantic
Figure 3.26. Example failure cases for our attribute manipulation framework, which are due to the visual quality of synthesized reference style image (top row) and failing of the photo style transfer method (bottom two rows).

layout of the input image and exhibit the desired attributes. Trained on our richly annotated ALS18K dataset, the proposed generative network can hallucinate many different attributes reasonably well and even allows edits with multiple attributes in a unified manner. For future work, we plan to extend our model’s functionality to perform local edits based on natural text queries, e.g. add or remove certain scene elements using referring expressions. Another interesting and more challenging research direction is to replace the proposed two-staged model with an architecture that can perform the manipulation in a single shot.
Figure 3.27. Attribute manipulation results obtained with our framework by using DPST [42] and FPST as the style transfer method. Given a natural scene image as input, we select a set of transient attributes and consequently obtain different images of the same scene reflecting the characteristics of these attributes.
Figure 3.28. Attribute manipulation results automatically generated by our framework by using DPST [42] and FPST [43] as the transferring mechanism. Given a natural scene image as input, we select a set of transient attributes and consequently obtain different images of the same scene reflecting the characteristics of these attributes.
4. LEARNING TO MANIPULATE PERSON VIDEOS WITH NATURAL LANGUAGE

In this section, we present a language-based video editing approach using garment descriptions for full-body person videos\textsuperscript{12}. We show that proposed method gives promising results and has high potential for video editing with natural language. Moreover, our method provides to learn permutation invariant set-based features thanks to approach being inspired by recently proposed Generative Query Networks [49].

4.1. Introduction

Humans are good at identifying a person in a scene from a natural language description. For instance, if we are given that the person is wearing “a white horizontal striped yellow t-shirt and blue denim jeans”, we do have a fairly good idea about what that such a person looks like independent from their pose. Similarly, given an image, one can easily describe the visual characteristics of a person in plain language. This interplay between vision and language suggests a close relationship between these modalities. In the last couple of years, researchers from computer vision and natural language processing fields have started to work on problems involving translation between these modalities. While generating textual descriptions from images, aka image captioning, is a well-studied topic in the integrated vision and language literature [135–137], there are only a few works which address its inverse problem, i.e. synthesizing images from natural language descriptions, which is known as text-to-image synthesis [38, 138–140]. Especially, benefiting from the progress of generative adversarial networks (GAN) [1], the researchers were able to obtain fairly good photo-realistic images for this problem as of the late. As a recently proposed task which is closely related to text-to-image synthesis, semantically manipulating images through natural language descriptions aims at modifying an input image in a way that the resulting image better aligns with the target text [38, 45–47]. This requires the corresponding models to not

\textsuperscript{12}This work will be presented in Sets & Partitions Workshop on Neural Information Processing Systems (WNeurIPS 2019).
In this work, we introduce a new task of manipulating person videos with natural language, which introduces new challenges for video synthesis. First, synthesizing the output video according to textual description of the target look requires handling the variations in the motion and pose information observed throughout the entire source sequence. More importantly, achieving photo-realism also strives to satisfy spatio-temporal consistency. To deal with these difficulties, we propose a new deep model composed of two coupled modules, a representation network and a translation network, as will be explained later. Our results illustrated in Fig. 4.1 demonstrates the effectiveness of our approach that it successfully modifies an input person video by changing only the parts of the person relevant to the natural description while preserving all the remaining parts intact without deteriorating the semantic and temporal coherency.

Our contributions can be listed as follows: (i) We collect a new video dataset which comprises of full-body images of individuals and the textual descriptions of the clothes they wear. (ii) We formulate a new conditional video-to-video translation task whose goal is to semantically manipulate the outfit of a person in a video clip according to a target look given in terms of a natural language description. (iii) We propose a deep video manipulation model which exploits multiple observations of a person to capture features relevant to the task and subsequently use them in modifying the source video according to the target description. (iv) Our analysis reveals that our proposed approach significantly outperforms the previous works that carry out frame-wise edits, and produce temporally more coherent and semantically more relevant results.

4.2. Related work

Deep generative models. Our proposed framework is built upon two of the most popular models for generative modeling, generative adversarial networks (GANs) [1] and variational autoencoders (VAEs) [2]. GANs consist of a generator and a discriminator, simultaneously
trained in a competitive manner. While the purpose of the generator is to produce realistic-looking images, the discriminator aims at distinguishing generated sample images from real ones. VAE is a latent variable model that provides an alternative tractable formulation. Researchers have shown that these two approaches can also be used to learn disentangled representations by tweaking their objectives, such as InfoGAN [141] and $\beta$-VAE [142]).

Another interesting line of research has focused on hybrid approaches that combine VAEs with GANs, e.g. ALI [143], BEGAN [144], IntroVAE [145], to name a few.

**Image-to-image translation.** Pix2pix [39] and CycleGAN [33] are the first examples of image-to-image translation models that are trained to transform an input image to a target domain. These models have been recently extended to multimodal [97] and multi-domain settings [98, 99] as well. Moreover, there are also some efforts to increase the realism and resolution of the outputs by considering multi-scale strategies [34, 40, 83].

**Language guided image synthesis/editing.** In text-to-image synthesis, the goal is to generate an image with a natural language description [38, 138–140]. On the other hand, semantic image manipulation aka language guided image editing models, such as SISGAN [45] and TAGAN [46], aim at modify a source image according to a textual description depicting the characteristics of the target image. As a person-oriented model, FashionGAN [47] differs from these generic models in that it considers a two stage GAN architecture which takes label segmentation maps and the binary attribute vectors of the input image as complementary information.
**Video synthesis and video-to-video translation.** The so-called video synthesis or video-to-video translation tasks are also related to our framework as they aim at generating temporally coherent video clips either from scratch [146–150], or according to a single image [151–153], or even by using a video from a different domain [154, 155].

### 4.3. Formulation

Given a video of a person and a target outfit description, we design our model to perform seamless and semantically meaningful edits on particular image regions to reflect what is being described in the text. Contrary to the previous works on editing person images (e.g. [47, 48]), we decided to rule out the need for additional prior information such as estimated body poses or segmentation of body parts or clothing. We believe that they put a strong constraint on the manipulation process and, moreover, they require additional data which may not be available or costly to obtain. Instead, in our framework, we exploit the nature of our source videos. Intuitively, some qualities of a person like hair color, facial features, skin tone, or the outfit are shared across all frames like whereas some other factors like pose vary from one frame to another. Note that, the target description only involves product information about a garment in terms of its color, texture and type, and we aim at replacing just the clothing according to the target text while preserving the look of remaining features, namely personal characteristics of the person and their pose. Formally, given a source video sequence $X_i = \{x_1^i, x_2^i, \ldots, x_n^i\}$, showing $n$ consecutive full-body images of a person $i$, and a target garment description $t$, our task is to synthesize an output video sequence $Y_i = \{y_1^i, y_2^i, \ldots, y_n^i\}$ by manipulating frames in $X_i$ in which the desired look portrayed in $t$ is transferred to the source person in a truthful and temporally coherent manner. See Fig. 4.1. for some examples.

At its core, our video manipulation model is partly motivated by recently proposed Generative Query Networks (GQN) [49], and consists of two subnetworks, a representation network $R$ and a translation network $T$, which are trained jointly in an end-to-end manner. In Fig. 4.2., we show a schematic illustration of our approach. The representation network
Figure 4.2. Schematic illustration of the proposed editing approach. (a) Source video sequence. (b) Representation network $R$ accepts multiple observations to compute a multi-view representation of the person by aggregating individually extracted representations. Translation network $T$ then uses this internal representation when manipulating input frames according to the target description.

takes in a set of randomly picked video frames from the source video and produces a multi-view representation $r$ that encodes person-specific features. This internal representation is then passed to the translation network as an extra condition in addition to the text embedding of the target description $t$. Hence, mathematically speaking, an output frame $y$ is computed via the mapping $T(x, r, t)$. For notational convenience, we omit person and frame subscripts whenever possible. We argue that accumulating information from multiple frames (observations) improves the manipulation quality by providing more evidence about which features should be preserved or forgotten. In the following section, we will detail each component of our framework.

4.4. Model Details

Our model can be seen as a special kind of conditioned VAE-GAN hybrid. For our representation network $R$, we utilize a $\beta$-VAE [142] with a spatial broadcast decoder [156] to obtain latent codes of individual frames, which are then aggregated to form the person representation. On top of that, we base our translation network $T$ on Pix2pixHD image-to-image translation model [40], in which we use adaptive instance normalization [157] to incorporate the condition information regarding both the target text description and the person-specific features extracted by $R$. To encode textual descriptions in a more aligned manner, we train a visual-semantic embedding model [158]. Moreover, we employ carefully
selected loss functions to further enforce temporal coherency on the overall manipulation results. Below, we introduce each component in more detail.

**Semantic Text Embedding.** We train a visual-semantic embedding network using the approach by Kiros et al. [158] to find a joint embedding space for images and text in our training data, in a similar fashion as done in previous work [45, 46]. This step can be interpreted as a pre-training step where visually grounded representations of texts are learned by optimizing a pairwise ranking loss that maximizes the similarity between embeddings of the matching image and text pairs while minimizing the similarity between the non-matching pairs. Refer to [158] for more details.

**Representation Network.** In our context, we can distinguish between two sources of information in a given input video: frame-specific features that smoothly vary across consecutive frames such as body or camera pose, and video-specific features which are consistently observed in the whole sequence encoding person-specific features (e.g. facial features, body type, hair color, clothing). The representation network’s task is to figure out which features are essentially important and must remain intact during the manipulation cycle. In that regard, the benefits of this network are two fold. First, by aggregating information from multiple frames, it provides a consistent representation of the person appearing in the source video and accordingly assists the translation network in manipulating the source frames according to the target text. Second, simultaneously training with the translation network leads to more disentangled representations, which is critical for the success of our framework.

Fig. 4.3.(a) shows the structure of the representation network $R$. Its main component is a $\beta$-VAE [142] with a spatial broadcast decoder [156], which spatially tiles the sampled latent code $z$ and concatenates fixed $x$ and $y$ coordinate channels (ranging from -1 to 1 horizontally and vertically), and then applies a stack of fully convolutional layers with $1 \times 1$ stride. The representation network randomly chooses $k$ frames from the input video containing $n$ frames ($k \leq n$), and uses the VAE to encode each observation. Finally, a permutation invariant, multi-view representation of the person is obtained by element-wise summing of individual
latent codes of the observations. A similar approach is also taken by Eslami et al. [49] in their own representation network to describe scenes. To alleviate the computation burden, the representation network works at half of the resolution of the translation network. To be more specific, it accepts 128 \times 96 color images as input. Our representation network is supervised directly by the reconstruction of the selected video frames, as shown below, and indirectly by joint training with the translation network, as explained later.

$$L_{VAE} = \frac{1}{k} \sum_{t=1}^{k} \mathbb{E}_{q_{\theta}(z'|x')} \left[ \log p_{\phi}(x'|z') \right] - \beta D_{KL} \left( q_{\theta}(z'|x') \parallel p(z') \right)$$

(27)

Our formulation encourages the VAE to learn disentangled features. However, it is important to note that some caution must be taken in identifying the person representation \(r\) to be passed on to the translation network. Since it will act as an auxiliary condition vector (in addition to the target text embedding), here we only want to transfer the features relevant to the manipulation process. We implement this by a simply masking strategy. Let \(m\) denote a binary mask vector of size equal to the dimension of \(z\). The representation \(r\) is computed as the non-zero components of the vector \(m \ast \sum \ z^i\) where \(\ast\) denotes element wise product. This simple strategy can be interpreted as a fixed hard selection mechanism to filter out redundant features for semantic manipulation. In our implementation, this operation reduces the dimensionality of latent codes by half.

**Translation Network.** We build our translation network \(T\) on Pix2pixHD [40]. As illustrated in Fig. 4.3.(b), it consists of three parts: An encoder, a series of residual blocks and a decoder. The input frame \(x\), which is of size 256 \times 192, is passed through convolutional layers in the encoder until it has been downsampled to obtain a 16 \times 12 \times 512 dimensional spatial feature map \(f_0\). In each residual block, features are first conditioned by the text embedding \(t\) and then the multi-view person representation \(r\) by using the adaptive instance normalization (AdaIN) [157]. The AdaIN operation can be interpreted as injecting a feature-wise linear modulation (FiLM) layer [159] after the instance normalization [160]. Hence, given input feature maps from residual block \(i\), output to the next residual block \(i + 1\) can be
Figure 4.3. Our video manipulation framework consists of two subnets: (a) representation network $R$ which is responsible for extracting relevant person-specific features from multiple frames, and (b) translation network $T$ which carries out the semantic editing process according to target text description by taking into account the extracted person representation.

defined by
\[
\hat{f}_i = \gamma^t \ast \frac{f_i - \mu(f_i)}{\sigma(f_i)} + \beta^t, \quad f_{i+1} = \gamma^r \ast \frac{f_i - \mu(f_i)}{\sigma(f_i)} + \beta^r
\]

where $\gamma^t, \beta^t$ and $\gamma^r, \beta^r$ are learned affine transformations for text and person-specific conditionings, respectively. After residual blocks, the last (conditioned) feature map is fed into the decoder, which consists of several convolutional transpose layers to upsample it to the original resolution to obtain the manipulated frame $y$. In the decoder, we also apply instance normalization in all convolutional transpose layers except the last layer. We use ReLU activation in all convolutional and convolutional transpose layers in all parts of the network.

To train our translation network we use an image-wise discriminator for measuring photorealism. Moreover, we add a perceptual loss for semantic consistency and a temporal smoothness term to further enforce temporal coherency. We detail these losses below.

Image-wise Discriminator. The GAN objective for the image-wise discriminator is defined as:
where $D$ denotes the multi-scale PatchGAN discriminator [40, 161], which is trained to distinguish the pair of a real frame and matching text $(x, t)$ from pairs of a real frame and mismatching text $(x, \hat{t})$ or manipulated (fake) image and relevant text $(y, \bar{t})$, $y = T(x, r, \bar{t})$, in a similar fashion as done in text-to-image synthesis works [38, 45]. Furthermore, as illustrated in Fig. 4.4, we incorporate the conditional information using a FiLM layer [159]. The idea resembles the projection discriminator proposed by [162].

**Perceptual Loss.** Our perceptual loss is quite standard. As done in [104], it tries to minimize $L1$ distance between the feature maps of the input frame $x$ and the manipulation result $y = T(x, r, \hat{t})$, extracted by a VGG-19 network [163] trained on ImageNet [164].

$$\mathcal{L}_{img}^{GAN}(T, D) = \mathbb{E}_{(x,t)}[\log D(x, t)]$$

$$+ \mathbb{E}_{(x,\hat{t})}[\log(1 - D(x, \hat{t}))]$$

$$+ \mathbb{E}_{(x,\bar{t},r)}[\log(1 - D(y))]$$

$$\mathcal{L}_{L1}^{img}(T) = \mathbb{E}_{(x,t,r)}[\|\Phi_{VGG}(x) - \Phi_{VGG}(y)\|_1]$$

**Temporal Smoothness.** Our frame-based architecture may cause temporal inconsistencies across synthesized video frames. To mitigate this, we employ an additional loss for temporal smoothness. We utilize pre-trained FlowNet2 model [165] to compute optical flows between consecutive frames, and then we use pre-computed optical flow to enforce two manipulated sequential frames to be temporally coherent. Like the perceptual loss, our loss for temporal
Smoothness is also based on feature matching, defined as:

\[
\mathcal{L}_{L_1}^{vid}(T) = \mathbb{E}_{(x^t, t, r)} \left[ \| \Phi_{VGG}(y^{t-1}) - \Phi_{VGG}(F(y^t)) \|_1 \right]
\]  

(31)

where \( F(y^t) \) denotes the warped frame with optical flow to the previous frame.

**Final Objective.** Our video manipulation framework’s final objective simultaneously optimizes the following objectives for jointly training the representation network \( R \) and the translation network \( T \):

\[
\mathcal{L}_T = \min_T \left( \max_D \mathcal{L}_{\text{GAN}}^{img}(T, D) + \lambda_1 \mathcal{L}_{L_1}^{img}(T) + \lambda_2 \mathcal{L}_{L_1}^{vid}(T) \right), \quad \mathcal{L}_R = \mathcal{L}_{\text{VAE}} + \lambda_3 \mathcal{L}_T
\]

(32)

where we empirically set the weights to \( \lambda_1 = \lambda_2 = 10 \) and \( \lambda_3 = 1 \) in our experiments.

### 4.5. Experiments

**Fashion Video Dataset.** We collected a new video dataset from an online shopping site, which contains short video clips of individuals wearing different kinds of garments. Each clip includes full-body images of a single person moving around a scene, showing how the clothing looks from different angles. Moreover, it is also provided with a textual product description of the garment, detailing its visual features (e.g. its color, its material properties or its design details) as well as its category (e.g. dress, jumpsuits, trousers, jumper, skirt, pant). After pre-processing, we obtained 3178 video clips out of which 2579 are used for training and 598 for testing. In terms of image counts, there are approximately 109K distinct video frames, which are split into about 88K training and 21K testing frames.

**Training Details.** In our experiments, we used Adam optimizer with \( \beta_1 = 0.5 \) and \( \beta_2 = 0.999 \) and the learning rate of 0.0002. We defined separate optimizers for \( D \), \( T \), and \( R \), and due to the GPU memory limitation, we set batch size to 8. For our ablation studies, we trained our models approximately 70 epochs. As our word embedding, we utilized pretrained Wikipedia 1M fastText model [166]. We set \( \beta = 4 \) for \( \beta \)-VAE and we picked \( k = 8 \) random video frames to obtain multiple observations. In the experiments, we resized the input frames.
to $128 \times 96$ pixels for $R$ and $256 \times 192$ pixels for $T$. Lastly, SISGAN and TAGAN is trained for 100 epochs as with their original settings.

**Results.** In Fig. 4.5., we provide qualitative comparisons for our approach and two frame-wise models, SISGAN [45] and TAGAN [46]. As can be seen, our model performs significantly better in terms of both photo-realism and semantic relevance to the target text. Note that it performs only the necessary edits to the input sequence. We carry out our quantitative analysis in two different aspects. First, we evaluate the quality in term of photo-realism via commonly used Inception Score (IS) [167], Fréchet Inception Distance (FID) [168], and the recently proposed Fréchet Video Distance (FVD) [169] for judging the quality over the whole output sequence. Secondly, we adapted the visual-semantic similarity (VS) measure, as suggested in [170], to assess the manipulation performance of the models according to the target natural language descriptions. VS measures the cosine similarity between the manipulated video frames and their corresponding target texts on the common semantic space learned in our pre-training phase. For a fair comparison, the outputs of our approach and TAGAN and the proposed are resized to $64 \times 64$ (the resolution of SISGAN) before estimating the evaluation scores. As shown in Table 4.1., our method gives considerably better results than the existing approaches in terms of both photo-realism and semantic relevance to the target descriptions.

### Table 4.1. Quantitative comparison against the state of the art models. Our approach outperforms the existing methods by a large margin in terms of all evaluation measures. For IS and VS, higher is better. For FID and VFD, lower is better.

<table>
<thead>
<tr>
<th>Method</th>
<th>IS</th>
<th>FID</th>
<th>VFD</th>
<th>VS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SISGAN [45]</td>
<td>2.33 ± 0.03</td>
<td>79.20</td>
<td>1674.75 ± 132.02</td>
<td>0.05 ± 0.13</td>
</tr>
<tr>
<td>TAGAN [46]</td>
<td>2.43 ± 0.03</td>
<td>22.13</td>
<td>828.16 ± 79.59</td>
<td>0.13 ± 0.15</td>
</tr>
<tr>
<td>Ours</td>
<td>2.72 ± 0.04</td>
<td>11.42</td>
<td>542.95 ± 84.32</td>
<td>0.13 ± 0.15</td>
</tr>
</tbody>
</table>

**Ablation Study.** We conduct an ablation study in order to analyze the contributions of temporal smoothness objective and the representation network. Please note that here we consider images of $256 \times 192$ pixels, which is the actual resolution of our translation network’s output. We can see in Table 4.2. that each of these components plays a crucial role...
<table>
<thead>
<tr>
<th>Target description</th>
<th>Video sequence (shortened)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>White Blouses</strong> Made from cotton. V-neckline Paper bag style. Long flared sleeves with pearl detail. Regular fit</td>
<td><img src="image1" alt="Video sequence" /></td>
</tr>
<tr>
<td><strong>Yellow Blouses</strong> Made from viscose Boat neckline Keyhole back with tie-up detail Long embroidered bell sleeves Regular fit</td>
<td><img src="image2" alt="Video sequence" /></td>
</tr>
<tr>
<td><strong>Blue Camis Vests</strong> Made from cotton spandex blend Lattice v neckline Sleeveless Regular fit</td>
<td><img src="image3" alt="Video sequence" /></td>
</tr>
</tbody>
</table>

Figure 4.5. Qualitative comparison against existing semantic image manipulation approaches.

in achieving better realism. Interestingly, the representation network contributes the most to the fidelity of our results. On the other hand, we find that including temporal smoothness has a negative effect on the VS score. We suspect that this may be related to inaccurately estimated optical flows used during the training with temporal smoothness, which let the model learn incorrect correlations regarding semantic manipulations.

**What Features Are Important for Manipulation?** We try to identify the characteristics
Table 4.2. Ablation study of the proposed semantic video manipulation framework.

<table>
<thead>
<tr>
<th>Method</th>
<th>IS</th>
<th>FID</th>
<th>FVD</th>
<th>VS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our Approach</td>
<td>3.33 ± 0.04</td>
<td>22.64</td>
<td>647.90 ± 94.81</td>
<td>0.24 ± 0.17</td>
</tr>
<tr>
<td>- Temporal Smoothness</td>
<td>3.05 ± 0.13</td>
<td>15.13</td>
<td>729.76 ± 75.61</td>
<td>0.34 ± 0.16</td>
</tr>
<tr>
<td>- Representation Network</td>
<td>2.99 ± 0.09</td>
<td>29.85</td>
<td>902.46 ± 105.94</td>
<td>0.36 ± 0.16</td>
</tr>
</tbody>
</table>

of person features that are picked by the translation network. For our analysis, we select a collection of 8 videos from the test set, which contains four different persons wearing two different outfits and in which each outfit has a similar one in terms of its textual description within this. Fig. 4.6. presents t-SNE [171] visualizations of the person images obtained with the latent embeddings of the β-VAE in our representation network $R$, the first 10 dimensions picked by translation ynetwork $T$, and the rest 10 dimensions, respectively. In Fig. 4.6.(a), we can see that β-VAE provides a clear separation between the individuals and the clothings. However, as we show in Fig. 4.6. (b) and (c), the first 10 and the last 10 dimensions of the latent space encode different properties. That is, while the former captures the pose of the persons well, the latter mainly represents the appearance of the outfits. This supports our intuition that joint training of the representation and translation networks helps the β-VAE model to learn relevant semantic features for manipulation and better disentangle the underlying factors of variations, as we detail next.

**Measuring Disentanglement.** Our dataset lacks the ground truth factors of variation, hence, we measure disentanglement by examining the latent traversals. In particular, we compare the β-VAE in our representation network $R$ and a normal β-VAE trained alone on the person images. In each row of Fig. 4.7., we show the person images reconstructed by traversing a single feature dimension of the latent space while keeping the values of the remaining dimensions fixed. We find that some of these dimensions encode sensible factors related to the input person videos such as azimuth, identity, garment type and color. Training β-VAE jointly with the translation network $T$ leads to better disentanglement. For example, the semantic dimension encoding the azimuth covers a large range of angles.
Figure 4.6. t-SNE visualizations of the person images obtained with (a) 20-dimensional embeddings learned by the $\beta$-VAE in our representation network $R$, (b) the first 10 dimensions that are passed to our translation network $T$, and (c) the last 10 dimensions (zoom in to see the details).
In this section, we introduced the new task of semantic manipulation of person videos using natural descriptions, together with a new dataset composed of videos of individuals and textual descriptions of their clothes. As a step towards solving this problem, we also proposed a deep semantic video manipulation model that learns to construct a person representation from multiple observations and exploit this internal representation while
performing necessary modifications to the source video according to the textual description of the target look. Our approach gives significantly better results than existing frame-based methods and performs semantically relevant and temporally coherent edits. As such, there are also a number of ways in which this work can be extended. Instead of employing a hard selection strategy, one may alternatively use a soft attention mechanism to select features from the representation network. In our work, we have avoided incorporating human segmentation maps but that information can also be easily integrated to our framework if needed.
5. CONCLUSION

In conclusion, we presented new data-driven approaches for alpha matting, visual attribute manipulation and language-based editing. Our rigorous experiments have shown that proposed approaches present better or competitive results with existing methods. Besides, our visual attribute manipulation method is the first sample of image editing using both attribute and layout conditioned image generation and photo style transfer and proposed editing tool is the first editing tool which enables to control desired transient attributes in continous manner by increasing or decreasing.

In the following, we discuss some potential applications and future directions for our methods to improve the results and simplicity.

For alpha matting, we showed that our sparse sampling method is also work with sparse user inputs since we do not make any spatial assumption while collecting color samples. Similarly, recent work SampleNet matting [23] show that learning to color sampling produce better results than directly estimating alpha matte. It may be valuable to anaylze the capability of deep learning methods for sparse user inputs as simple mouse clicks or scribbles and build a self-supervised learning model to utilize random scribble and mouse clicks to learn the required color samples with very simple user interactions.

Our visual attribute manipulation method can be used to augment realistic data in various weather, season and time conditions to improve the robustness of other computer vision tasks such as semantic scene segmentation, computer vision based self-driving cars, etc. Recently, semantic image synthesis is greatly improved by new semantic layout conditioning solutions [172, 173]. In this regard, better image synthesis approaches will improve the our results in terms of color quality and fine details. Similary, new photo style transfer methods [44, 174] have been proposed to increase quality and lower the running time. Nevertheless, we observe that existing photo style transfer methods still produce some visual artifacts and need post-processing. Moreover, they are not capable of transferring some visual attributes such as fog and clouds. Because such attributes need to insert new materials to the
input image unlike color and illumination transfer. As mentioned before, another interesting
and more challenging research direction is to replace the proposed two-staged model with an
architecture that can perform the manipulation in a single shot. In that way, we may transfer
specific attributes such as clouds and fog more successfully.

Language-based image editing is getting more attention in computer vision and machine
learning community. Because, giving machines the ability to imagine possible new objects
or scenes from linguistic descriptions and produce their realistic renderings is arguably one
of the most challenging problems in computer vision and artificial intelligence research.
Recent advances in deep generative models have led to new approaches that give promising
results towards achieving this goal. There are many aspects to improve our results from
better text embedding to permutation invariant feature learning. For this purpose, one needs
to develop new network architecture, loss function and conditioning mechanism for better
disentanglement of variation of visual information and to provide photorealistic visual quality.
REFERENCES


[38] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In Proceedings


Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In


Thesis Title / Topic: LEARNING BASED IMAGE AND VIDEO EDITING

According to the originality report obtained by myself, my thesis advisor by using the Turnitin plagiarism detection software and by applying the filtering options stated below on 25/12/2019 for the total of 158 pages including the a) Title Page, b) Introduction, c) Main Chapters, d) Conclusion sections of my thesis entitled as above, the similarity index of my thesis is 5.4%.

Filtering options applied:
1. Bibliography/Works Cited excluded
2. Quotes excluded
3. Match size up to 5 words excluded

I declare that I have carefully read Hacettepe University Graduate School of Science and Engineering Guidelines for Obtaining and Using Thesis Originality Reports; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

Name Surname: Levent KARACAN
Student No: N13249103
Department: Computer Engineering
Program: Computer Engineering
Status: □ Masters □ Ph.D. □ Integrated Ph.D.

Date and Signature

ADVISOR APPROVAL

APPROVED.

Assoc. Prof. Dr. M. Erkut ERDEM

(Title, Name Surname, Signature)
CURRICULUM VITAE

Credentials
Name, Surname : Levent KARACAN
Place of Birth : 16/05/1987
Marital Status : Married
E-mail : karacan@cs.hacettepe.edu.tr
Address : Hacettepe University, Beytepe Campus, Computer Engineering Dept.

Education
B.Sc. : Computer Engineering, Erciyes University, Kayseri, Turkey
M.Sc. : Computer Engineering, Hacettepe University, Ankara, Turkey

Foreign Languages
English

Work Experience
Research Assistant at Hacettepe University, 2012-2018
Research Assistant at Iskenderun Technical University, 2019

Areas of Experiences
Image Processing, Computer Vision, Machine Learning
Publications

“Manipulating Attributes of Natural Scenes via Hallucination”,
ACM Transactions on Graphics(TOG), 2019, Volume 39, Issue 1
Levent Karacan, Zeynep Akata, Aykut Erdem and Erkut Erdem

“Image synthesis in multi-contrast MRI with conditional generative adversarial networks”,
IEEE Transactions on Medical Imaging, 2019, Volume: 38 , Issue: 10, Pages: 2375 - 2388
Salman UH Dar, Mahmut Yurt, Levent Karacan, Aykut Erdem, Erkut Erdem, Tolga Çukur

“Alpha matting with kl-divergence-based sparse sampling”,
IEEE Transactions on Image Processing(TIP), 2017, Volume: 26 , Issue: 9, Pages: 4523 - 4536
Levent Karacan, Aykut Erdem, Erkut Erdem

“Structure-preserving image smoothing via region covariances”,
Levent Karacan, Erkut Erdem, Aykut Erdem

Conference and Workshop Papers

“Manipulating Person Videos with Natural Language”
Levent Karacan, Mehmet Günel, Aykut Erdem and Erkut Erdem

“Finding location of a photograph with deep learning”
Yunus Emre Özkoş, Tarik Yilioğlu, Levent Karacan and Aykut Erdem
Signal Processing and Communications Applications Conference(SIU 2018)

“Adjusting transient attributes of outdoor images using generative adversarial networks”
Levent Karacan, Aykut Erdem and Erkut Erdem
Signal Processing and Communications Applications Conference(SIU 2017)
“Image matting with KL-divergence based sparse sampling”
Levent Karacan, Aykut Erdem and Erkut Erdem
IEEE International Conference on Computer Vision (ICCV 2015)

“Image colorization via dense correspondences”
Mehmet Gunel, Levent Karacan, Aykut Erdem, Erkut Erdem
Signal Processing and Communications Applications Conference (SIU 2014)

“2-opt based artificial bee colony algorithm for solving traveling salesman problem”
Bahriye Akay, Emre Aydogan, Levent Karacan
World Conference on Information Technology (WCIT 2011)

Preprints
“Learning to generate images of outdoor scenes from attributes and semantic layouts”
Levent Karacan, Zeynep Akata, Aykut Erdem and Erkut Erdem