Reiko Lettmoden
r.lettmoden@tu-bs.de

Referee Prof. Dr.-Ing. Martin Eisemann
eisemann@cg.tu-bs.de
Computer Graphics Lab, TU Braunschweig

CLIPasso: Semantically-Aware Object Sketching

Yael Vinker, Ehsan Pajouheshgar, Jessica Y. Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, Ariel Shamir

Seminar

May 22, 2023

Computer Graphics Lab, TU Braunschweig

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe.

Braunschweig, 22. Mai 2023

Lettmoden

Abstract

This paper reviews the work of Vinker et al. "CLIPasso: Semantically-Aware Object Sketching" which is a paper about synthesizing a sketch from an image. Preliminaries from the domain of neural networks are summarized to understand the contribution of Vinker et al. Their proposed work is compared with other works of the sketching domain, own examples are generated and the scope of their contribution is discussed.

Contents

Chapter 1

Introduction

Sketches are a minimal visual representation made of strokes portraying a scene or object of interest. The key element is drawing semantic features, which describe the characteristics of an object class while placing the strokes spatially consistent to the reference object instance. In computer vision, the task is to synthesize a sketch from an image while keeping the sketch simple, e.g. as abstract as possible. Recent state-of-the-art relies on sketching data sets and thus is either restricted to object classes seen during training or is unable to abstract features well.

Vinker et al. [VPB+22] present a novel method based on a trained CLIP model by Radford et al. [RKH+21] which provides good results in multiple classification tasks without requiring training on the data sets. Therefore, they employ a differentiable pipeline to generate sketches from Bézier curves and optimize the curve parameters based on the desired level of abstraction. Layer activations of the CLIP model are used as perceptual loss and attention maps in combination with an edge map are used to initialize stroke positions.

In user studies, Vinker et al. compare instance-recognition and category level recognition rates. For instance-recognition, multiple images of an object class are sampled and sketches have to be assigned to its corresponding image. For category level the correct class has to be predicted by a human or artificial classifier such as a neural network. They achieve a slightly worse instance-recognition rates compared to previous state-of-the-art since their sketches are generated with a higher level of abstraction. Vinker et al. show a slightly better performance of category level recognition rates, as they achieve 91% with a ViT-B/32[DBK+21] classifier compared to 77% accuracy of previous sketching methods.

This paper is structured as followed: first, neural networks in computer vision are explained in chapter 2 to understand CLIP. Vinker et al.'s method is explained in chapter 3. Experiments in comparison to related works and own experiments are shown in chapter 4. Limitations and contributions of

Vinker et al.'s work are discussed and summarized in chapter 5.

Chapter 2

Preliminaries

Neural Networks are widely used in computer vision tasks since they are able to learn a latent feature space from the highly complex image space by training with a lot of data on a task. A neural network (NN) often consists of multiple billion parameters which are optimized during training to perform a task such as classification, which is to assign a class label to a given image. Commonly, the parameters are optimized during training with data and its associated labels (supervised learning), e.g. images and its class labels in the classification task. Based on the task, the NN predicts a label of an input image during training and a loss is calculated based on the distance to the true label of the image. Then, parameters are optimized with gradient descent by backpropagating the loss through the layers of the network. Among used architectures in computer vision are convolutional neural networks (CNNs) and more recently vision transformers.

Convolutional Neural Networks are using convolutional filters, whose parameters are learned during training. Convolutional filters are stacked onto multiple layers and thus imply a local and hierarchical bias [ZF14]. This means activations rely on local information from the previous layer and cover a larger area on the input image with more depth. Zeiler et al. [ZF14] show that convolutional neural networks (CNNs) learn more abstract concepts with more depth, e.g. edge and color filters are learned in the first layers while class properties are learned in later layers.

He et al. [HZRS16] introduce Residual CNNs (ResNets) with shortcut connections which map the identity between two layers. Thus, layer output plus the input of a previous layer are added up which allows ResNets to learn the identity. Hence, ResNets with more layers and less parameters are introduced which achieved new state-of-the-art performance.

Transformer are introduced by Vaswani et al. [VSP+17] and implement a attention based encoder-decoder architecture. Attention is implemented

similar to a key-value store and achieves state-of-the-art in the field of natural language processing (NLP). A query is fed into the self-attention block, attends highest to a key and thus its value is obtained. In the process, key, value and query are linearly projected by weight matrices which are learned during training.

Dosovitskiy et al. [DBK+21] propose a vision transformer (ViT) which splits an image into 16x16 patches on which self-attention is applied. They show that ViT exhibit less inductive bias than CNNs since self-attention is applied globally in comparison to the locality and hierarchy bias of convolutions. Thus, a ViT outperforms the previously dominating ResNets given sufficient pre-training [DBK+21].

CLIP Radford et al. [RKH+21] merge NLP with computer vision to train a model capable of generalizing well on unseen data sets. Therefore, image and text pairings are scraped online, for example an image of a dog with the subtitle "Pepper the aussie dog". Then, the image and text pairs are mapped into feature space by a vision and text encoder. The model is trained with contrastive learning, e.g. the distance in feature space of corresponding text and image pairs are minimized while the distance to the other pairs in the batch is maximized.

CLIP is trained with both a ResNet and ViT vision encoder on 400 million image and text pairs. The largest ViT model takes 12 days on 256 V100 GPUs to train and the largest ResNet model 18 days on 592 V100 GPUs [1]. This approach improves zero-shot learning meaning it can generalize better on unseen data sets in comparison to previous supervised approaches [RKH+21].

[1] For comparison, the ImageNet 1k [RDS+15] classification data set only contains 1.2 million images with class labels only(no text) and a ResNet50 can be trained on 4 V100 GPUs in 55 hours [WTJ21].

Chapter 3

Method

3.1 Overview

From a given input image the task is to generate a sketch which generates a good abstraction of the image. Vinker et al. [VPB+22] utilize Bézier curves with 4 control points (tuples of x and y coordinates) to draw sketches. The first and last point indicate the start and end point of a curve, whereas the other two other points control the curve's course. This allows curves with a loop whereas a Bézier curve with 2 control points is a line and and 3 control points allow only curvature into one direction. Initial positions of the curves are initialized with attention maps based on CLIP[RKH+21] and edge maps of the input image, see section 3.2.

The number of Bézier curves is chosen based on the desired level of abstraction, e.g. with less curves the level of abstraction increases. A differentiable rasterizer [LLGRK20] produces an image from the parameters of the Bézier curves. Since the rasterizer is differentiable the loss can be back-propagated and the curve parameters optimized iteratively. To measure the similarity of the generated sketch with regards to the image, a geometric and semantic loss based on CLIP[RKH+21] is employed, as discussed in section 3.3.

3.2 Stroke initialization

Since the image space and thus parameter space of the curves is high dimensional, optimization results depend a lot on the initialization of stroke parameters. Hence, Vinker et al. [VPB+22] propose a novel initialization based on attention maps from a trained CLIP[RKH+21] ViT image encoder.

Attention maps highlight semantically important image regions in an attention layer. The average attention map from all layers is computed which they denote to as saliency map. Additionally, they extract an edge map with XDoG[WKO12], e.g. they highlight edges (large gradients) in an

image. The edge map is multiplied with the saliency map to get regions of interest from which initialization points are sampled.

Different initialization methods are visualized in figure 3.1 provided by Vinker et al. Random initialization commonly causes strokes to cluster on contours and thus miss important semantic features. Initialization with ViT saliency maps show both attention on semantic features such as the face of the cat and non-semantic features such as the corners of the image. Hence, strokes are initialized in the face and outside of the cat leading to less contours of the body. Combining ViT saliency maps with XDog results in points sampled only from the cats body since attention values outside edges are canceled out. This approach is able to distribute initial strokes better and thus creates better abstractions.

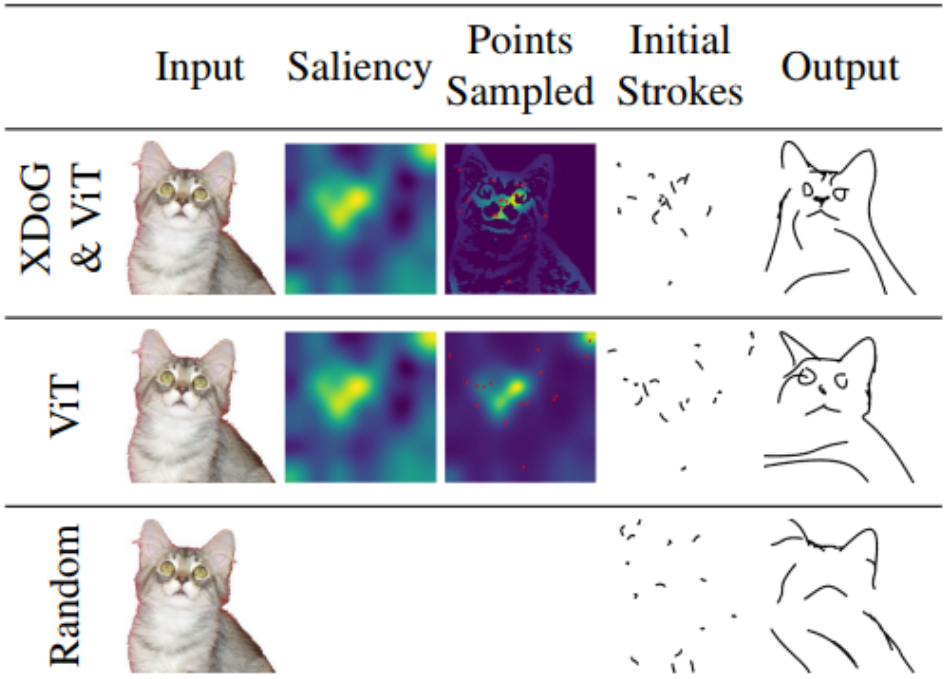

Figure 3.1: Comparison of different initialization methods by Vinker et al. [VPB+22]

3.3 Geometric and semantic loss

The ability to conceptualize an image means to extract its geometric and semantic information. Geometric consistency describes spatial consistency of the generated sketch and reference image, e.g. the contours of the scene. Additionally, semantic consistency describes outlining object properties which

characterize object classes, such as cat ears etc. Depending on the level of abstraction the semantic loss plays a major role as visual features have to be more discriminative. Vinker et al. utilize the ResNet CLIP model to employ a novel geometric and semantic loss.

Figure 3.2 visualizes optimization with regards to different layers of the ResNet CLIP model. While the first two layers are not capturing semantic and relevant geometric information, layers 3 and 4 focus on contours of the body and semantic features such as facial properties. Layers 5 and the fully connected layer on the other side are less geometrically consistent and focus on semantic features, such as the nose, eyes or other facial features. Hence, Vinker et al. calculate the geometric loss from the L2 distance of the third and forth convolutional layers and semantic loss from the last fully connected layer. Both the semantic and geometric loss are added with the semantic loss being weighted down by 0.1.

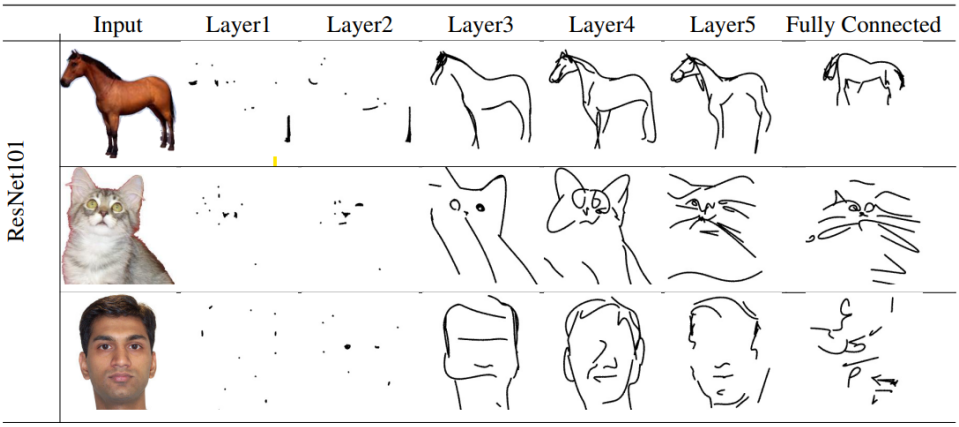

Figure 3.2: Optimization with regards to layers of the CLIP[RKH+21] ResNet[HZRS16] model provided by Vinker et al. [VPB+22].

Chapter 4

Experiments

To bring the contribution of Vinker et al. into context, sketching examples are compared with other works in section 4.1. Some examples of my own are shown in section 4.2.

4.1 Comparison to other works

Figure 4.1 shows a sketch of a teacup in comparison to other works. The works of (A)[KP20] and (B)[LSHG17] are not able to capture geometric features to make the teapots distinguishable but are able to capture the semantics of the teapot. CLIPDraw[FSW21] is not geometrically consistent at all while (C)[LLM+19] is able to capture many geometric features. However, (C) is not able to create different semantic abstractions and as a consequence less abstract sketches compared to Vinker et al. Thus, CLIPasso shows less geometric consistency compared to (C).

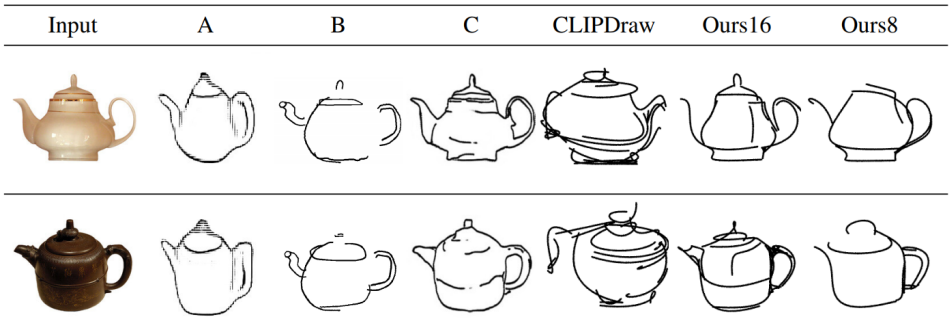

Figure 4.1: Comparison to other works by Vinker et al. [VPB+22]. Left to right: (A) Kampelmühler and Pinz [KP20], (B) Li et al. [LSHG17], (C) Li et al. [LLM+19], CLIPDraw [FSW21] and CLIPassso with 16 and 8 strokes.

4.2 Examples

Figure 4.2 shows some generated sketches of my own. It can be seen that the mask heavily influences the sketch quality. Utilizing no masks causes the loss to focus on the whole pictures such that some strokes stay in the background. The U2-net[QZH+20] can generate a mask of the wrong objects or not the whole object of interest as clusters of strokes are appearing. Generating a mask with human supervision causes the best results, probably due to better initialization of strokes and since the loss is only optimized with regards to the object of interest.

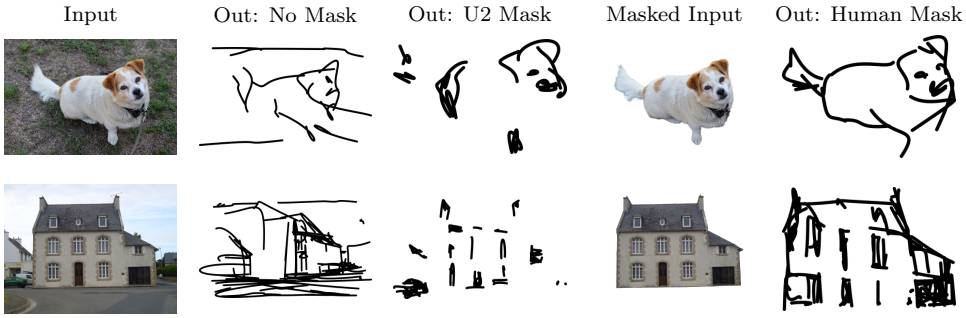

Figure 4.2: Sketches generated by CLIPasso[VPB+22] with no mask, mask generated by a U2-net[QZH+20] and masked input with human supervision. For the dog 16 strokes are used while for the house 64 strokes are used. The upper input image of the dog is published by the user "Drow male" under the CC BY-SA 4.0 license. The image of the house is provided by the public domain.

Chapter 5

Conclusion and Discussion

5.1 Discussion

Vinker et al. contribution is applying CLIP to a new task which exhibits great zero-shot capabilities. Since data driven approaches require data sets and are limited to the style and categories of the data set the authors are not relying on collecting data which costs money and time.

While previous data driven methods in the sketching domain have to be trained and are inferred once during testing, sketch curves are optimized through a CLIP based loss iteratively. Optimization takes 2000 iterations for 3 different seeds results in 18 minutes computation time on a V100 GPU which is not a consumer GPU. My experiments were performed on a Nvidia T4 GPU, which is approximately comparable to a consumer GeForce GTX 1080 Ti GPU, and took approximately 23 minutes for 3 seeds. Thus, CLIPasso still has to be optimized as it is computational and time expensive due to the CLIP vision encoder forward pass for each optimization iteration. However, Vinker et al. show that choosing a more light-weight ResNet50 instead of ResNet101 decreases the quality of generated sketches. Here, the dependency on CLIP can be evaluated as a disadvantage since training CLIP on more recent vision architectures is hardly possible because a huge GPU cluster (expensive!) and data set would be required.

Utilizing layer activations as similarity measure for a query image to a reference is commonly done in computer vision. This is done through perceptual losses in image synthesis[ZIE+18] or geometric and semantic losses in image retrieval[CAS20]. However, Vinker et al. provide detailed ablation studies for their model choices and how to utilize CLIP layers best with regards to the sketching problem.

Additionally, Vinker et al. slightly discuss architecture bias of both vision transformer and convolutional neural networks. Examining use cases of both networks in ablation studies brings reasonable explanations for their decisions and how to extract information from images. Vinker et al. state,

that the ViT focuses on geometric consistency while the ResNet101 achieves a good geometric and semantic balance. However, the used ViT is already outperformed by the Swin Transformer which restricts attention to local windows and employs a hierarchical architecture similar to CNNs. Thus, a further study of more recent CNN and visual transformer architectures would be interesting for future work.

Their proposed method is focusing on object sketching but also generalizes well on sketching whole images due to the employed losses. For object sketching it depends on masked images which are generated with an U2-net[QZH+20] from raw images. Some experiments of my own show, that the masking often fails since the mask only captures a part of the object of interest if at all. To get the best results, there is still human interaction required to manually mask the object of interest.

5.2 Summary

Compared to previous work, Vinker et al.[VPB+22] utilize a pretrained CLIP[RKH+21] model with great generalizability and thus is not relying on a sketching data set. Their major contribution is deploying a CLIP based saliency map for stroke initialization and applying CLIP losses to create semantically and geometrically consistent object sketches. Optimizing Bézier curves iteratively allows the usage of multiple stroke styles and multiple levels of abstraction. However, the optimization is computationally expensive and the best results require human supervision to mask the object of interest.

Bibliography

[CAS20] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying deep local and global features for image search. In *European Conference on Computer Vision*, pages 726–743. Springer, 2020.

[DBK+21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[FSW21] Kevin Frans, Lisa B Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *arXiv preprint arXiv:2106.14843*, 2021.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[KP20] Moritz Kampelmuhler and Axel Pinz. Synthesizing human-like sketches from natural images using a conditional convolutional decoder. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3203–3211, 2020.

[LLGRK20] Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.

[LLM+19] Mengtian Li, Zhe Lin, Radomir Mech, Ersin Yumer, and Deva Ramanan. Photo-sketching: Inferring contour drawings from images. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1403–1412. IEEE, 2019.

[LSHG17] Yi Li, Yi-Zhe Song, Timothy M Hospedales, and Shaogang Gong. Free-hand sketch synthesis with deformable stroke models. *International Journal of Computer Vision*, 122(1):169–190, 2017.

[QZH+20] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern recognition*, 106:107404, 2020.

[RDS+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[RKH+21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 8748–8763, 2021.

[VPB+22] Yael Vinker, Ehsan Pajouheshgar, Jessica Y. Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *ACM Trans. Graph.*, 41, 2022.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WKO12] Holger Winnemöller, Jan Eric Kyprianidis, and Sven C Olsen. Xdog: An extended difference-of-gaussians compendium including advanced image stylization. *Computers & Graphics*, 36(6):740–753, 2012.

[WTJ21] Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021.

[ZF14] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[ZIE+18] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.