https://doi.org/10.7840/kics.2023.48.12.1577

Deep Learning-Based Image Stitching Technique Using Transformer

Byungchan Choi[◆], Seungwon Shin[*],
Jihyun Kim[**], Sehwan An[**], Jihan Joo[**],
Haewoon Nam[°]

ABSTRACT

Image stitching is an image processing technique that combines multiple overlapping images into one. Classical image stitching extracts common feature keypoints from input images and use them as reference points for aligning the images. Convolutional neural network-based image stitching network combines latent feature vectors to produce stitched output from input overlapping images. This paper proposes a transformer-based image stitching network along with its implementation and training strategies. Unlike previous methods, the proposed transformer-based image stitching network explicitly learns to produce stitched output images from connection between patches of input images. It achieves 3.7dB higher PSNR and 0.12 higher SSIM than classical image stitching technique.

Key Words : Image Stitching, Deep Learning, Transformer

Ⅰ. Introduction

Classical image stitching techniques use matching feature keypoints from input overlapping images as reference points to combine them into one[1]. Feature extraction algorithms, such as Scale-Inavriant Feautre Transform (SIFT), are used to acquire feature keypoints from input images[2]. Random Sampling Consensus (RANSAC) is used to find matching features and compute the homography parameters that can minimize the difference between overlapping images[3]. Homography parameters from RANSAC are used to align overlapping images into one. Classical techniques require fine tuning on feature extraction, feature matching, and RANSAC. Since RANSAC conducts random sampling on input feature keypoints, it produces different outputs at each use. Depending on the number of features extracted from input images, RANSAC requires different thresholds to produce proper output images. Therefore, classical image stitching algorithms cannot achieve generalized performance.

Deep learning-based image stitching techniques train a deep learning network to directly produce stitched images from input images with overlapping parts. Convolutional Neural Network (CNN) is often used to extract local features from input images. Latent vectors from CNN layers are combined into one to produce stitched outputs. Since this method does not create direct connections between features, CNN-based image stitching network implicitly learns to stitch overlapping images. Its performance depends on how input images are overlapped, which leads to decreased generalization.

This paper proposes the use of transformer for deep learning-based image stitching. The contribution of this paper is as follows:

- Transformer-based network is proposed and implemented for image stitching.
- Training strategies, such as loss function setup and dataset setup, are proposed to train transformer-based image stitching network to learn image stitching from connections between patches of input images.

※ This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2023-RS-2023-00258639) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).

◆ First Author : LIG Nex1, RF Seeker R&D, and Hanyang University, byungchan.choi@lignex1.co.kr, 연구원, 정회원
° Corresponding Author : Hanyang University, Division of Electrical Engineering, hnam@hanyang.ac.kr, 정교수, 정회원
* Hanyang University, Division of Electrical Engineering, fusr0da@naver.com, 학생(학부과정)
** LIG Nex1, RF Seeker R&D, jihyun.kim@lignex1.co.kr; sehwan.an@lignex1.co.kr; jihan.joo@lignex1.co.kr

논문번호 : 202308-069-C-LU, Received August 30, 2023; Revised September 14, 2023; Accepted September 14, 2023

Ⅱ. Transformer-based Image Stitching Network

Fig 1 is transformer-based image stitching network proposed by this paper. It is trained to produce groundtruth stitched images from two overlapping images. It receives two overlapping images, Source A and Source B, as its inputs. As proposed by vision transformer, the input images are divided into patch embedding vectors[4]. Groundtruth stitched images are also divided into patch embedding vectors and inserted to the transformer as target sequence for the transformer's decoder. Same CNN layer with kernel size 8x8 and stride 8 produces 8x8 image patch embedding vectors for both input and groundtruth images. Groundtruth patch embedding vectors are appended with Start-ofSentence (SOS) token at the front. SOS token is randomly generated at the beginning of the training. It is used to indicate the beginning of input image patch embeddings sequence during transformer's training and inference. Groundtruth patch embeddings appended with SOS token are inserted to the transformer's decoder as target input sequence along with additive target mask sequence. Transformer's output embedding vectors are converted into patches and assembled as an output image.

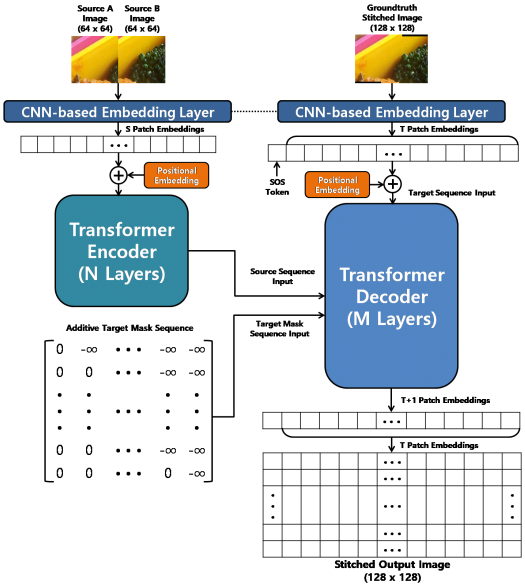

Fig. 1. Transformer-based Image Stitching Network

Each output embedding vector from transformer is the outcome from the connections between all the patches of input source images. While learning to produce groundtruth stitched images, the transformer will find which input image patch overlaps with others. This will be used to stitch two input images by constructing overlapping parts of groundtruth images.

$$l_n = \begin{cases} 0.5(x_n - y_n)^2, & \text{if } |x_n - y_n| < delta \\ delta \times (|x_n - y_n| - 0.5 \times delta), & \text{otherwise} \end{cases} \tag{1}$$

Image stitching task requires high pixel level accuracy in order to precisely stitch overlapping parts of input images and produce high quality stitched outcome with strong details. Transformer has high level of flexibility and freedom in training due to its low inductive bias. In order for the transformer to learn image stitching task and reach convergence during training, it needs the loss function that can guide it to produce high pixel level precision. Pixel-by-pixel L2 loss can be used to achieve high pixel level accuracy. However, L2 loss is sensitive to outliers, which can lead to failure during training. This paper utilizes Huber loss from (1) in order to achieve high pixel level accuracy and minimize the effect of outliers during training.

Ⅲ. Training and Experiment Results

Transformer-based image stitching network from Fig 1 is implemented with PyTorch 1.13[5]. It is trained on Nvidia GPU by Adam Optimizer with learning late = $1e^{-4}$, β_1 = 0.9, β_2 = 0.999, and ε = $1e^{-8}$. It is trained for 70 epochs.

15000 training images, 5000 validation images, and 1000 test images are generated from MS-COCO dataset[6]. A random section of each image in MS-COCO dataset is cropped as Source A image. Source B image is randomly cropped next to Source A image while maintaining minimum overlap ratio of 30%. Fig 2 shows how each input and groundtruth data is organized.

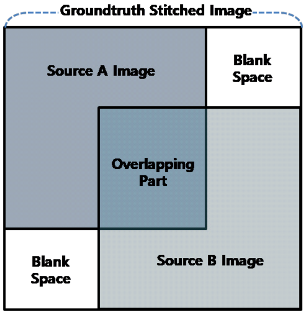

Fig. 2. Groundtruth Setup

Performance of the proposed transformer-based image stitching network is evaluated on two metrics: Peak Signal-to-Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM)[7]. Table 1 shows the network's performance on training and validation dataset. It is organized according to overlap ratio between Source A image and Source B image. Performance metrics on training dataset and validation dataset from Table 1 are similar, which indicates that there is no overfitting in the proposed deep learning network structure and training setup.

Table 2 shows image stitching performance between classical method and the proposed method on test dataset. Classical method is a image stitching pipeline with SIFT and RANSAC implemented in OpenCV[8,9]. According to Table 2, the proposed method achieves higher PSNR and SSIM than classical method in all overlap ratio conditions.

Fig 3 shows output samples generated from classical method and the proposed method. Sample 1 and 2 show the cases when stitched outputs are similar in both classical method and the proposed method Sample 3 and 4 show that classical method produces stitched output image with shape distortions along outer edges even if overlap ratio is high between source images. However, from source images of Sample 3 and 4, the proposed method produces stitched output image with strong similarity in structure and details to groundtruth images. Along with results from Table 2, Fig 3 demonstrates that the proposed method is a viable method for image stitching applications.

Table 1. Training Result

	Average PSNR [dB]		Average SSIM	
Overlap Ratio	Training	Validation	Training	Validation
30 ~45%	23.74	23.53	0.75	0.75
45 ~60%	24.15	24.20	0.77	0.78
60 ~75%	24.86	24.75	0.80	0.80
Over 75%	26.62	26.60	0.81	0.81
Total Average	24.66	24.58	0.78	0.78

Table 2. Image Stitching Comparison

	Classical Method[8] SIFT + RANSAC		Proposed Method	
Overlap Ratio	Average PSNR [dB]	SSIM	Average PSNR [dB]	SSIM
30 ~45%	21.24	0.68	23.73	0.75
45 ~60%	21.06	0.70	24.26	0.77
60 ~75%	21.93	0.68	25.01	0.80
Over 75%	20.04	0.60	27.03	0.82
Total Average	21.10	0.66	24.80 (+3.7)	0.78 (+0.12)

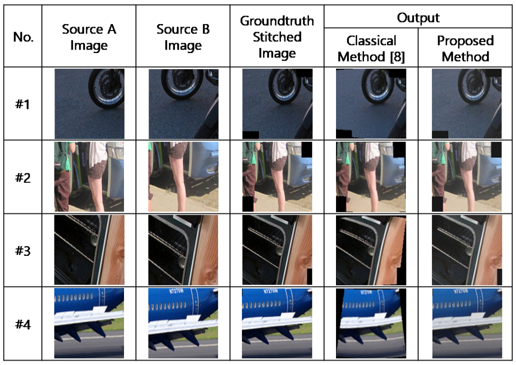

Fig. 3. Stitched Output Samples Test Set

Ⅳ. Conclusion

This paper shows the implementation and training strategies for transformer-based image stitching network. This network learns to stitch input overlapping images based on the connections between all the input image patches. The proposed method achieves higher performance in PSNR and SSIM when compared to classical image stitching pipeline[8].

The proposed method can be further improved by training the network to achieve perceptual similarity from VGG loss[10]. With this loss, the proposed transformer-based image stitching network can possess the capability to capture details and styles during training. With these improvements, it can be tested on dataset with more challenging homography conditions.

References

[1] M. M. Hossain, H. Lee, and J. Lee, "Fast image stitching for video stabilization using sift feature points," *J. KICS*, vol. 39, no. 10, pp. 957-966, 2014. (https://doi.org/10.7840/Kics.2014.39C.10.957).

[2] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. Seventh IEEE Int. Conf. Computer Vision*, vol. 2, pp. 1150-1157, 1999. (https://doi.org/10.1109/ICCV.1999.790410).

[3] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381-395, 1981. (https:// doi.org/10.1145/358669.358692).

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. (https:// doi.org/10.48550/arXiv.2010.11929).

[5] A. Paszke, S. Gross, F. Massa, et al., "Pytorch: An imperative style, high-performance deep learning library," *Advances in NIPS*, vol. 32, 2019.

[6] T.-Y. Lin, M. Maire, S. Belongie, et al., "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th Eur. Conf.,* Zurich, Switzerland, Sep. 2014. Proc. Part V 13, Springer, pp. 740-755, 2014. (https://doi.org/10.1007/978-3-319-10602-1_48).

[7] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600-612, 2004. (https://doi.org/10.1109/TIP.2003.819861).

[8] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *Int. J. Computer Vision*, vol. 74, pp. 59-73, 2007. (https://doi.org/10.1007/s11263-006-0002-3).

[9] G. Bradski, "The OpenCV library," *Dr. Dobb's J. Software Tools*, 2000.

[10] C. Ledig, L. Theis, F. Huszár, et al., "Photorealistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. CVPR*, pp. 4681-4690, 2017. (https://doi.org/10.1109/CVPR.2017.19).