ICIC Express Letters
Volume **15**, Number **9**, September **2021**

ICIC International ©2021 ISSN 1881-803X
pp. 1007–1012

DEEP CORRELATION BASED HOMOGRAPHY ESTIMATION FOR IMAGE STITCHING

Joshua Santoso[1,3], Williem[2,3] and Rini Wongso[3,*]

[1]Department of Information and Communication Engineering
Inha University
100 Inha-ro, Michuhol-gu, Incheon 22212, Korea
joshuajano@yahoo.com

[2]Verihubs
South Jakarta 12950, Indonesia
williem@verihubs.com

[3]Computer Science Department
School of Computer Science
Bina Nusantara University
Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
[*]Corresponding author: rwongso@binus.ac.id

Received November 2020; accepted February 2021

Abstract. *Homography estimation is essential in many image transformation cases. Currently, there are two different paradigms, specifically non-learning-based and learning-based methods. However, both methods are still prone to erroneous homography estimation. To address this problem, we introduce a deep correlation based homography estimation that provides more stable and accurate results. The main novelty of this paper is to consider the correlation for each extracted feature, which contributes to removing redundant features and increasing the accuracy. Experimental results show the usability of the proposed method to produce more reliable stitched images. The proposed method achieves state-of-the-art results compared to the previous methods.*

Keywords: Homography estimation, Correlation layer, Image stitching, Image transformation

1. **Introduction.** The homography is a promising potential technology in image transformation owing to its ability to transform one image plane from one camera into a different camera view by changing the position and the camera rotation, respectively. It is well-known that homography is prominent for stitching images and creating panoramas. Nowadays, many applications such as augmented reality, virtual reality, google-street-view, medical imaging, satellite mapping, and unmanned aerial vehicle (UAV) [1] are applying homography as the foundation method.

To estimate the homography matrix, there are two different approaches, specifically the traditional method and deep network method. Szeliski [2] introduces two categories namely direct photometric-based and sparse feature-based. Direct methods are robust to images with textureless but are trouble large motion. Contrariwise, feature-based methods are robust to large motion but heavily rely on feature correspondences. Feature-based separates the process into two processes: the first estimate local features using SIFT (*scale invariant feature transform*) [3], and the last is feature matching. However, fallacy can occur during feature matching. To address this problem, some works make use of optimization algorithms such as RANSAC (*random sample consensus*) [4] to minimize the error.

DOI: 10.24507/icicel.15.09.1007

Recent deep networks achieve state-of-the-art postulate in several cases such as image classification, pose estimation [5] and human body reconstruction. In homography estimation task, Tone et al. [6] introduce a novel deep homography estimation network to compute a homography between two images. On the other hand, Sampetoding et al. present the combination of convolutional neural network (CNN) based on image transformation for image stitching [7] and show a promising result for image stitching task.

This paper investigates the problem of developing a deep network method for homography estimation that is applied for image stitching. First, we generate dataset for the training purpose of deep homography estimation for image stitching. Second, we introduce a matching network for computing the correlation for each extracted feature. The features correlation is useful to guide the network finding the overlapping regions. Finally, we apply the estimated homography matrix to performing image stitching between two images. Our experiments show that our method can produce a promising image for image stitching cases and achieve the state-of-the-art performance.

2. **System Model and Methods.** Figure 1 shows an overview of the proposed method. The network receives two images as the input: left image I_L and right image I_R. Note that this method assumes that f_L has similar characteristics of f_R and vice versa. $F_{LR} \in (F_L \cap F_R)$. Then, a feature extractor is performed on each image. The correlation layer finds the overlapping information between the features. Finally, the homography matrix is estimated using 4-point parameterization. The details are described in the following subsections.

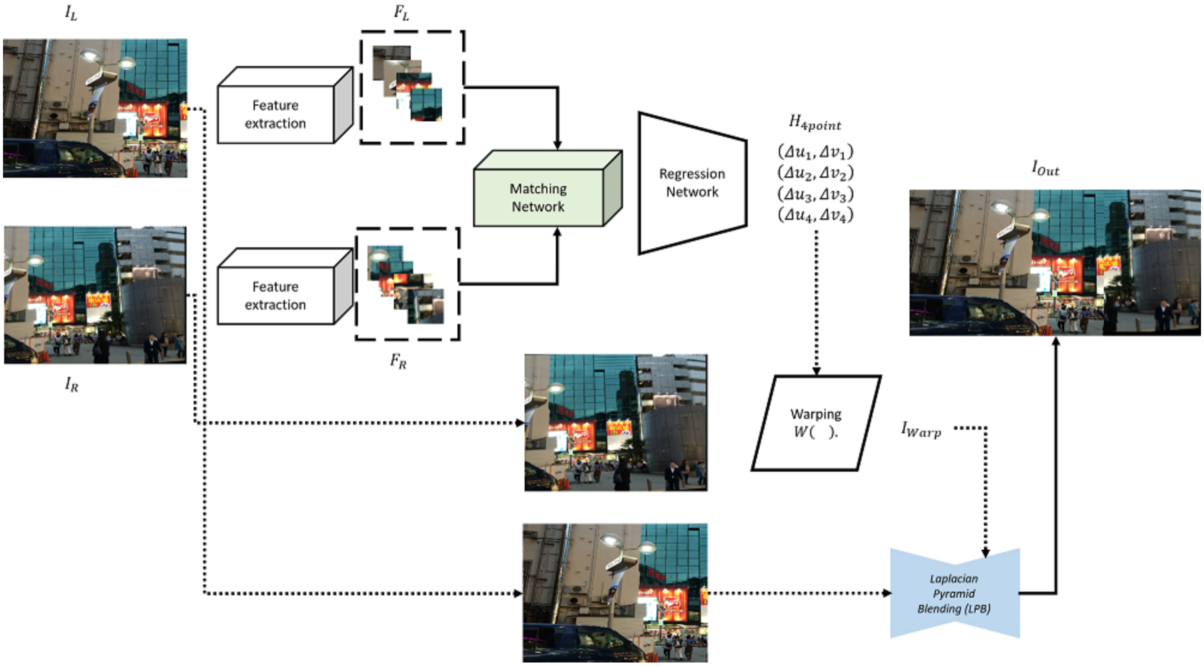
I_L
F_L
Feature extraction
Matching Network
Regression Network
H_{4point}
$(\Delta u_1, \Delta v_1)$
$(\Delta u_2, \Delta v_2)$
$(\Delta u_3, \Delta v_3)$
$(\Delta u_4, \Delta v_4)$
I_{Out}
Feature extraction
I_R
F_R
Warping
$W(\ )$.
I_{Warp}
Laplacian Pyramid Blending (LPB)

FIGURE 1. Our framework is divided into three subtasks and gets H_{4point} as the output. The goal is utillizing H_{4point} to stitch the image.

2.1. **Feature extraction.** The first stage of the pipeline is feature extraction. We utilize *DenseNet121* [8] originally trained on ImageNet for the task of image classification. As an initial step, we feed a left part image I_L and a right part image I_R as inputs. Each input produces the feature map $f \in \mathbb{R}^{h \times w \times d}$ information. f can be defined as dense spatial grid with w as width and h as height d-dimensional local descriptors.

2.2. **Matching network.** As the primary role of calculating homography estimation, the matching network role is to find the correlation between each feature. The outputs from foregoing process are left image features f_L and right image features f_R where $f_L, f_R \in \mathbb{R}^{h\times w\times d}$. Furthermore, matching network attempts to find the correlation map $c_{LR} \in \mathbb{R}^{h\times w\times (h\times w)}$ whereupon comprises all pairwise semblance between each feature $\mathbf{f}_L \in f_L$ and $\mathbf{f}_R \in f_R$. With the intention of foraging c_{LR}, NCC method [9] is exploited as described in Equation (1).

$$NCC(\mathbf{f}_L, \mathbf{f}_R) = \frac{\sum_n \mathbf{f}_{L_n} \times \mathbf{f}_{R_n}}{\sqrt{\sum_n (\mathbf{f}_{L_n})^2 \times \sum_n (\mathbf{f}_{R_n})^2}} \tag{1}$$

By means of Equation (1), c_{LR} can be calculated at particular spatial location (i, j) as shown in Equation (2).

$$c_{LR}(i,j) = NCC(\mathbf{f}_L(i,j), \mathbf{f}_R(i,j)) \tag{2}$$

2.3. **Regression network.** The regression network produces 4-point parameterization H_{4point} as shown in Equation (3). The H_{4point} represents the corner displacement location. Please follow [6] for more detail about 4-point parameterization.

$$H_{4point} = \begin{bmatrix} \Delta u_1 & \Delta v_1 \\ \Delta u_2 & \Delta v_2 \\ \Delta u_3 & \Delta v_3 \\ \Delta u_4 & \Delta v_4 \end{bmatrix} \tag{3}$$

As the loss function, Euclidean distance is utilized as the primary loss function, as shown in Equation (4).

$$L_{L2} = \sum_{i=1}^{n} (y_i - f(x_i))^2 \tag{4}$$

2.4. **Warping and blending.** To get the homography transformation matrix H_M from H_{4point} and ground truth H_S, DCH utilizes *getPerspectiveTransform()* from OpenCV.

$$H_M = getPerspectiveTransform(H_S, H_{4point}) \tag{5}$$

Subsequently, by using H_M wrap the I_{Right} to get the warpped image I_{Warp} where (u^1, v^1) represents deformed image and (u, v) for original image location.

$$\begin{bmatrix} u^1 \\ v^1 \\ 1 \end{bmatrix} = H_M \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{6}$$

Later, I_L and I_{Warp} are stitched using Laplacian Pyramid Blending and produce the stitched image I_{Out} as shown in Equation (7).

$$I_{Out} = LPB(I_L, I_{Warp}) \tag{7}$$

2.5. **Dataset generation.** To generate the dataset, we utilize the concept from [6]. Given a large image I, first, we randomly crop a rectangle patch with size $(320, 240)$ width and height, respectively, at position p. This cropped image is I_L. Then, based on p information, we shift the location with δp while the range is $[10, 200]$. By considering the above information, we are able to crop the next input, which contains some similar features with I_L. By applying $p + \delta p$, we can get the new crop coordinate, which produces the other crop image I_R. We also calculate the homography transformation value by utilizing *getPerspectiveTransform()* by OpenCV and save the δp as a ground truth.

3. **Result.** Here we give a quick description of the datasets we use for training and evaluation. We collect the datasets from inria dataset [13], paris dataset [14], and tokyo-street-view dataset [12] and get 8000 images. The collected datasets are divided into 7000 and 1000 images for training and evaluation, respectively. As a comparison, we compared our method with RANSAC [4], DLT (direct linear transform) [11], DGM (deep geometry matching) [10], and DHN (deep homography network) [6].

There are two metrics for the comparison: pixel-signal-to-noise-ratio (PSNR) and mean-absolute-error (MAE) of position. For the PSNR comparison, we compare the stitched image result from each algorithm as visualized in Table 2. As shown in Figure 3, our method can produce more plausible results for each test image. The mean-absolute-error (MAE) of position is measured by calculating the distance between predicted 4-points and ground truth 4-points. Table 1 shows that our method can achieve the lowest position error compared to others.

Compared to existing CNN-based methods, our method outperforms them in terms of PSNR and position error. The idea of DHN [6] is to combine descriptors across images by concatenating descriptors along the channel dimension and utilize a single network. However, this method inflicts the network to do many tasks such as feature extraction,

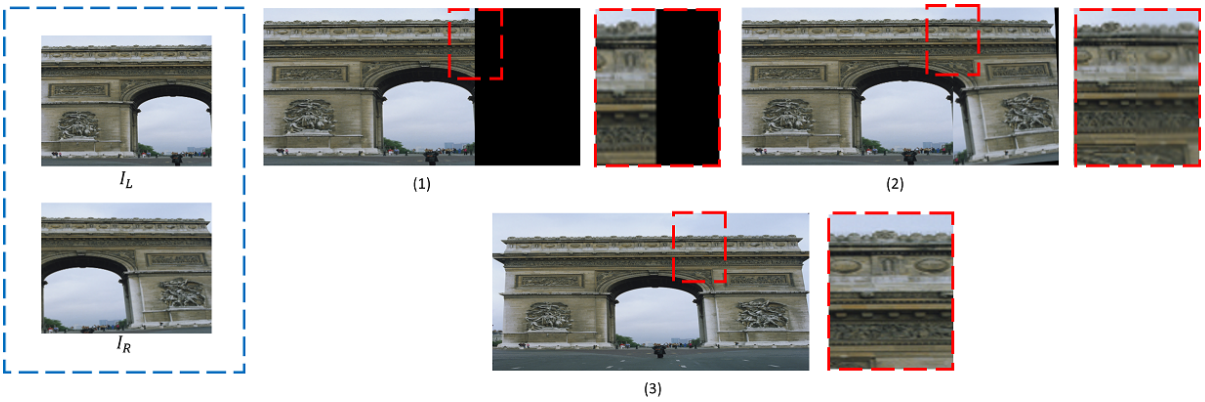

Figure 2. Visual comparison between RANSAC [4] and our proposed method. (1) is the result from RANSAC, (2) is ours and (3) is the ground truth.

Table 1. Comparison of mean position error

Method	Position error
DGM [10]	18.61
RANSAC [4]	28.63
DLT [11]	183.90
DHN [6]	12.32
DCH	8.23

Table 2. Comparison of PSNR

Method	PSNR
DGM [10]	17.41
RANSAC [4]	22.06
DLT [11]	14.40
DHN [6]	18.60
DCH	20.13

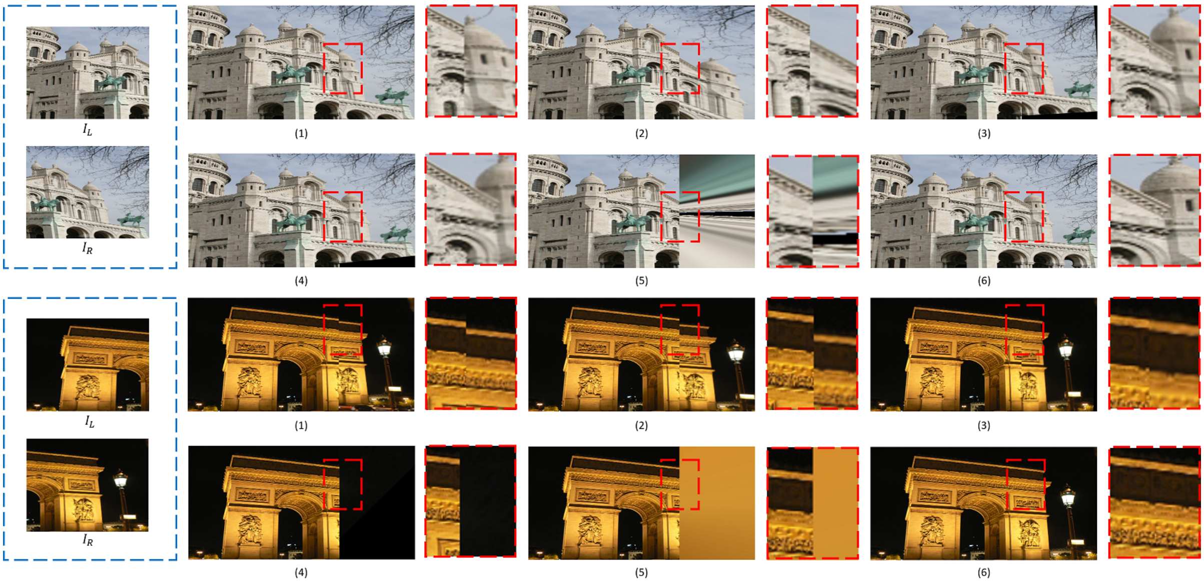

Figure 3. Visual comparison against the other methods. **From (1) to (6):** DHN [6], DGM [10], our proposed method (DCH), RANSAC [4], DLT [11], and ground truth.

feature matching, and regression inside a single network. This limitation has been solved by Rocco et al. [10]. This method splits each task into a small network. In detail, the feature extraction network is provided for each input respectively. Note each network shares its parameter. We analyze that most of the time, the shared parameter tends to be noise information to the others. As the result, the network suffers to estimate the accurate output.

Though RANSAC based method achieves higher PSNR value, RANSAC [4] method is heavily dependent on the feature extraction result. The images shown in the second column of Figure 3 and Figure 2 are examples of textureless images. In this case, RANSAC fails to produce a plausible result. Nonetheless, our method is still able to produce a promising result.

4. **Discussion and Conclusions.** This paper introduced a deep correlation based homography network (DCH) which utilized a correlation layer to improve the homography estimation result. The features for each image were extracted using the extraction network and matched using the correlation layer. Then, the 4-point parameterization of homography matrix was estimated using regression network. The correlation layer improved the accuracy of the homography parameters because it searched for the similar features as shown in the experimental results.

To train the network, this paper gathered a large image dataset of image pair (left and right). The network was trained and tested on the dataset. Note that, the proposed method was used to perform image stitching between two consecutive images. In the future, this method can be extended by incorporating depth information.

REFERENCES

[1] J. Liu, X. Qin, B. Qi and X. Cui, 3D online path planning of UAV based on improved differential evolution and model predictive control, *International Journal of Innovative Computing, Information and Control*, vol.16, no.1, pp.315-329, 2020.

[2] R. Szeliski, Image alignment and stitching, *Handbook of Mathematical Models in Computer Vision*, pp.273-292, 2006.

[3] D. G. Lowe, Object recognition from local scale-invariant features, *International Conference on Computer Vision*, vol.2, pp.1150-1157, 1999.

[4] Y. Zhao, R. Hong, J. Jiang, J. Wen and H. Zhan, Image matching by fast random sample consensus, *International Conference on Internet Multimedia Computing and Service*, pp.159-162, 2013.

[5] H.-Y. Lin, C.-C. Chang and S.-C. Liang, 3D pose estimation using genetic-based iterative closest point algorithm, *International Journal of Innovative Computing, Information and Control*, vol.14, no.2, pp.537-547, 2018.

[6] D. D. Tone, T. Malisiewicz and A. Rabinovich, Deep image homography estimation, *arXiv.org*, arXiv: 1606.03798, 2016.

[7] J. Sampetoding, B. Satriyawibowo, Williem, R. Wongso and F. Luwinda, Automatic field-of-view expansion using deep features and image stitching, *Procedia Computer Science*, pp.657-662, 2018.

[8] G. Huang, Z. Liu, L. V. D. Maaten and K. Q. Weinberger, Densely connected convolutional networks, *Conference on Computer Vision and Pattern Recognition*, pp.2261-2269, 2017.

[9] L. di Stefano, S. Mattoccia and M. Mola, An efficient algorithm for exhaustive template matching based on normalized cross correlation, *International Conference on Image Analysis and Processing*, pp.322-327, 2003.

[10] I. Rocco, R. Arandjelovic and J. Sivic, Convolutional neural network architecture for geometric matching, *Conference on Computer Vision and Pattern Recognition*, pp.39-48, 2017.

[11] Y. I. Abdel-Aziz, H. M. Karara and M. Hauck, Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry, *Photogrammetric Engineering and Remote Sensing*, pp.103-107, 1971.

[12] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi and T. Pajdla, 24/7 place recognition by view synthesis, *Trans. Pattern Anal. Mach. Intell.*, pp.257-271, 2018.

[13] H. Jégou, M. Douze and C. Schmid, Hamming embedding and weak geometric consistency for large scale image search, *Proc. of European Conference on Computer Vision*, 2008.

[14] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell and A. A. Efros, Context encoders: Feature learning by inpainting, *Conference on Computer Vision and Pattern Recognition*, 2016.