A MULTI-ANGLE PEDESTRIAN IMAGE PAIRING METHOD BASED ON PERSPECTIVE TRANSFORMATION

Sun, Shanqian
Department of Communication Design Science, Kyushu University

Inoue, Kohei
Department of Communication Design Science, Kyushu University

出版情報：ICIC Express Letters. 18 (12), pp.1257-1264, 2024-12. ICIC International
バージョン：

ICIC Express Letters
Volume **18**, Number **12**, December **2024**
 ISSN 1881-803X
pp. **1257–1264**

A MULTI-ANGLE PEDESTRIAN IMAGE PAIRING METHOD BASED ON PERSPECTIVE TRANSFORMATION

Shanqian Sun and Kohei Inoue

Department of Communication Design Science
Kyushu University
4-9-1 Shiobaru, Minamiku, Fukuoka 815-8540, Japan
sun.shanqian@kyudai.jp; k-inoue@design.kyushu-u.ac.jp

Received February 2024; accepted May 2024

Abstract. *Pedestrian re-identification is to match a person's identity across different cameras or locations in a video or image sequence. It involves detecting pedestrians from a frame sequence and using features such as human faces and clothes to identify and match. However, the features captured by only one camera are not robust enough with rotation problems. So, two cameras capture multi-angle images as they provide more information than single-angle images. This paper proposes a foreground segmentation method to divide the foreground into pedestrians. Then, a perspective transformation-based pairing method is proposed to match the same pedestrian's images captured by two cameras at the same place. The results demonstrate that the foreground segmentation method deals with only more than two classes that can be classified as problems. The pairing method can accurately pair the same person's images from different-angle cameras.*
Keywords: Pedestrian re-identification, Multi-angle, Image matching, Foreground segmentation

1. **Introduction.** Pedestrian re-identification aims to match pedestrians from cameras in different locations. Human face [1], gait [2], fingerprints [3], and other biological characteristics are utilized to distinguish pedestrians. However, all the characteristics are not robust to occlusion in a natural complex environment [5]. In our former study [4], we proposed a new HSV color space to indicate a person's characteristics and used the color histogram to match pedestrians. The result also showed that it works well only on images in the same direction.

To address the problem above, we plan to lead to direction information and only match the pedestrians in the same direction. Specifically, we first detect the foreground (moving part) in a frame sequence and then divide it into pedestrians. Then, we detect and utilize the direction information to match pedestrians. In our former research, we proposed a discrete Fourier transform-based foreground detection method [6] and an Otsu method-based foreground detection method. In this paper, we developed a foreground segmentation and pairing method. We should note that we use two cameras at the same place to capture different-angle images and use the pairing process to bind the images to the same person.

Section 2 discusses the main ideas and the drawbacks of three existing foreground segmentation methods. Section 3 proposes our improved foreground segmentation method and deduces the perspective transformation. Section 4 shows the experimental results, and Section 5 concludes this paper.

DOI: 10.24507/icicel.18.12.1257

2. **Related Work.** The k-means method is the most popular clustering method [7]. Here are some indexes to determine the k-value.

The first one is called the elbow method [8]. It calculates the sum of the squared errors (SSE), which is given by

$$SSE = \sum_{i=1}^{k} \sum_{p \in C_i} (p - m_i)^2 \tag{1}$$

where i is the number of clusters, p is the pixels in cluster i, and m_i is the centroid of cluster i.

The degree of clustering increases as the k-value grows. The SSE decreases rapidly when the k-value exceeds the number of clusters. However, it lowers slowly when the k-value exceeds the number of clusters. The main idea of this method is to find such an elbow to determine the k-value.

The second one is called the silhouette coefficient [9]. It measures how similar a data point is to its cluster compared to others. For a data point $i \in C_I$, let

$$a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i, j) \tag{2}$$

be the mean distance between i and all other data points in the same cluster, where C_I is the number of points belonging to cluster C_I, and $d(i, j)$ is the distance between i and j. $a(i)$ is a measure of how well i is assigned to its cluster, and the smaller the value, the better the assignment. We can also define

$$b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i, j) \tag{3}$$

to be the smallest mean distance of i to all points in any other cluster. The bigger the value, the better the assignment. Now, we can define the silhouette coefficient as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \text{ if } |C_I| > 1 \tag{4}$$

and

$$s(i) = 0, \text{ if } |C_I| = 1 \tag{5}$$

These can also be written as

$$s(i) = \begin{cases} 1 - \dfrac{a(i)}{b(i)}, & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ \dfrac{b(i)}{a(i)} - 1, & \text{if } a(i) > b(i) \end{cases} \tag{6}$$

and it is clear that $-1 \leq s(i) \leq 1$. And the $s(i)$ close to 1 means that the data is appropriately clustered, and if the $s(i)$ is close to -1, i would be more appropriate if it was clustered in its neighboring cluster. The optimal k-value corresponds to the biggest one.

The third one is called the Calinski-Harabasz index [10]. Like the silhouette coefficient, the CHI is defined as the ratio of the between-cluster separation to the within-cluster dispersion:

$$CHI = \frac{BCSS/(k-1)}{WCSS/(n-k)} \tag{7}$$

in which the $BCSS$ is the weighted sum of squared distances between each cluster centroid and the overall data centroid:

$$BCSS = \sum_{i=1}^{k} n_i ||c_i - c||^2 \tag{8}$$

where n_i is the number of points in cluster C_i, c_i is the centroid of C_i and c is the overall data centroid. The *WCSS* is the sum of squared distances between the data points and their respective cluster centroids:

$$WCSS = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - c_i||^2 \tag{9}$$

where x are the points in cluster C_i. The bigger the value, the better the assignment.

Finally, it is the Davies-Bouldin index [11]. Like the two methods above, it calculates the ratio of the within-cluster distance to the between-cluster distance. It is defined as

$$DBI = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{S_i + S_j}{d(c_i, c_j)} \right) \tag{10}$$

where S_i is the mean distance between the points in cluster i to their centroid c_i and defined as

$$S_i = \frac{1}{C_i} \sum_{x \in C_i} d(x, c_i) \tag{11}$$

and the smaller, the better.

In summary, the indexes above have some drawbacks. Firstly, clearly defining the elbow inflection point in the elbow method is problematic. Then, the remaining three indexes can be calculated only when the number of clusters exceeds 2, while the actual clusters could be only 1.

3. New Method.

3.1. Foreground segmentation.

To solve the problems above, we proposed a new method enlightened by the elbow method. As Equation (12) shows, it calculates the mean distance in all the clusters, in which $|C_i|$ is the number of points in cluster i, $d(x, c_i)$ is the distance between the points in cluster i and its centroid. The distances would decrease when the k-value increases. Then, it adds the distances together and selects the k-value corresponding to the smallest one to balance the number of clusters and distances.

$$D_k = \sum_{i=1}^{k} \left(\frac{1}{|C_i|} \sum_{x \in C_i} d(x, c_i) \right) \tag{12}$$

3.2. Foreground pairing.

This section talks about the segmented foreground pairing method. Perspective transformation can be used for point matching on two flats and can be written as the equation below [12, 13]:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} A_{2\times2} & T_{2\times1} \\ P_{1\times2} & s \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = H_{3\times3} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{13}$$

where transformation matrix $H_{3\times3}$ can be regarded as the combination of affine transformation matrix $A_{2\times2}$, translation transformation matrix $T_{2\times1}$ and the projection transformation matrix $P_{1\times2}$, s is a scale factor. Then, we can get the following equation from Equation (13):

$$\begin{aligned} x' &= \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \\ y' &= \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \end{aligned} \tag{14}$$

Then, by moving the denominators to the left and reform, we can get

$$\begin{aligned} xh_{11} + yh_{12} + h_{13} - x'xh_{31} - x'yh_{32} - x'h_{33} = 0 \\ xh_{21} + yh_{22} + h_{23} - y'xh_{31} - y'yh_{32} - y'h_{33} = 0 \end{aligned} \tag{15}$$

Finally, we get the matrix form as

$$\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x' \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \tag{16}$$

And, we can find the result does not change when H changes to aH as the equations below.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = aH \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{17}$$

$$\begin{aligned} x' &= \frac{ah_{11}x + ah_{12}y + ah_{13}}{ah_{31}x + ah_{32}y + ah_{33}} = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \\ y' &= \frac{ah_{21}x + ah_{22}y + ah_{23}}{ah_{31}x + ah_{32}y + ah_{33}} = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \end{aligned} \tag{18}$$

So, let $a = \frac{1}{h_{33}}$, we can get a new transformation matrix H' as

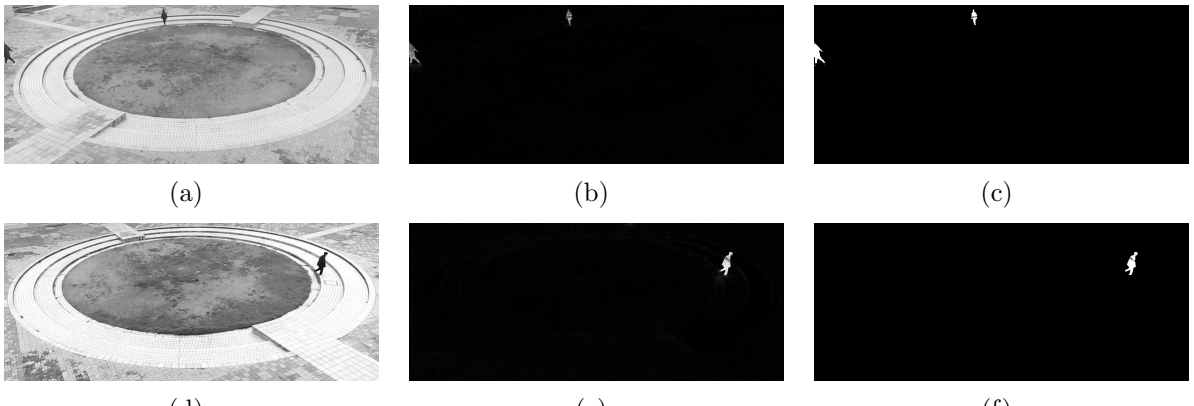
$$H' = aH = \begin{bmatrix} \frac{h_{11}}{h_{33}} & \frac{h_{12}}{h_{33}} & \frac{h_{13}}{h_{33}} \\ \frac{h_{21}}{h_{33}} & \frac{h_{22}}{h_{33}} & \frac{h_{23}}{h_{33}} \\ \frac{h_{31}}{h_{33}} & \frac{h_{32}}{h_{33}} & \frac{h_{33}}{h_{33}} \end{bmatrix} = \begin{bmatrix} h'_{11} & h'_{12} & h'_{13} \\ h'_{21} & h'_{22} & h'_{23} \\ h'_{31} & h'_{32} & 1 \end{bmatrix} \tag{19}$$

the degree of freedom is 8, and four-point pairs can determine the H'.

4. **Experimental Result.**

4.1. **Foreground segmentation.**

Figure 1 shows the detected foreground with our former research.



(a) (b) (c)

(d) (e) (f)

FIGURE 1. Foreground detection with our former research

(a) and (d) are the original frames captured by two cameras simultaneously. Then, we subtracted the background, acquired by a median filter, from the original frame to get the foreground, as (b) and (e) show. Finally, we used an Otsu method-based thresholding method for binarization, as (c) and (f) show. The dataset was taken at Kyushu University in Japan.

Figure 2 shows the relationship between the SSE and the number of clusters of (c) and (f) in Figure 1. The SSE of (c) decreases rapidly when $k < 2$ and decreases slowly when $k > 2$. So, $k = 2$ should be determined as the optimal k-value. However, (f) decreases slack, and finding an inflection point to determine the k-value is hard.

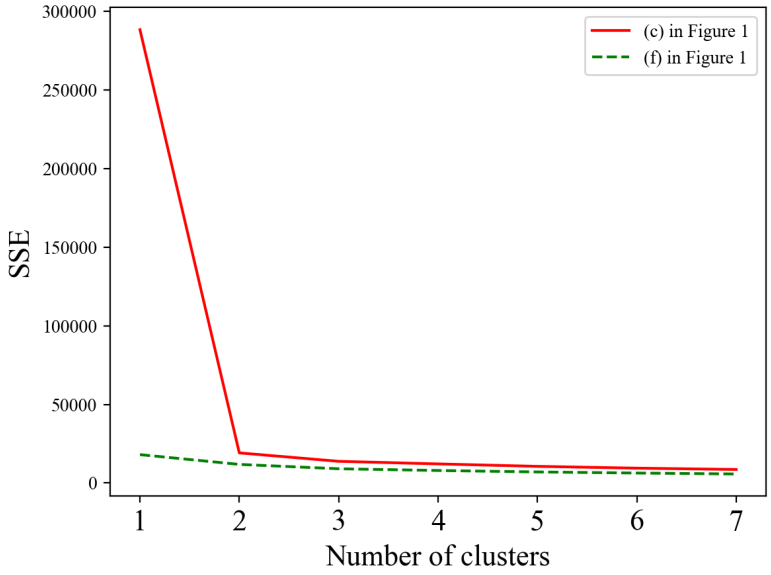


FIGURE 2. Connection between SSE and the number of clusters

Table 1 shows the calculation result of (c) in Figure 1 with the three remaining indexes and our new index. The suggested k-values correspond to the bold. The result shows most indexes can determine the right k-value. It should be noted that the $k = 1$ blanks of the three indexes are n/a, which means they are not applicable.

TABLE 1. Different indexes and selected k-value of (c)

Index	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
S. C.	n/a*	**0.961**	0.645	0.505	0.429	0.431	0.429
C. H. I.	n/a*	280962	**286439**	241651	245196	247234	246213
D. B. I.	n/a*	**0.053**	0.497	0.712	0.806	0.784	0.797
Ours	208.7	**25.7**	29.9	32.5	37.7	40.5	42.6

* These indexes need at least 2 clusters.

Table 2 shows the result of (f). Only our method can determine the correct k-value while the number of clusters is 1.

Figure 3 shows some foreground segmentation results with the determined k-value acquired by our index. (a), (b) and (c) are from one camera, and (d), (e), and (f) are from the other camera. (a) and (d) separate the foreground correctly, but the middle lower person in (e) is divided into two people. Moreover, three persons in (c) are recognized as two as two are too close to each other.

4.2. **Foreground pairing.** As the theory talked about in Section 3.2, we used four parallel points to calculate the perspective transform matrix like the black spots in (a) and

TABLE 2. Different indexes and selected k-value of (f)

Index	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
S. C.	n/a*	**0.483**	0.455	0.383	0.407	0.399	0.412
C. H. I.	n/a*	1666	**1903**	1746	1745	1757	1840
D. B. I.	n/a*	**0.731**	0.755	0.921	0.812	0.857	0.738
Ours	**16.4**	21.5	24.4	28.9	30.9	34.0	35.3

* These indexes need at least 2 clusters.

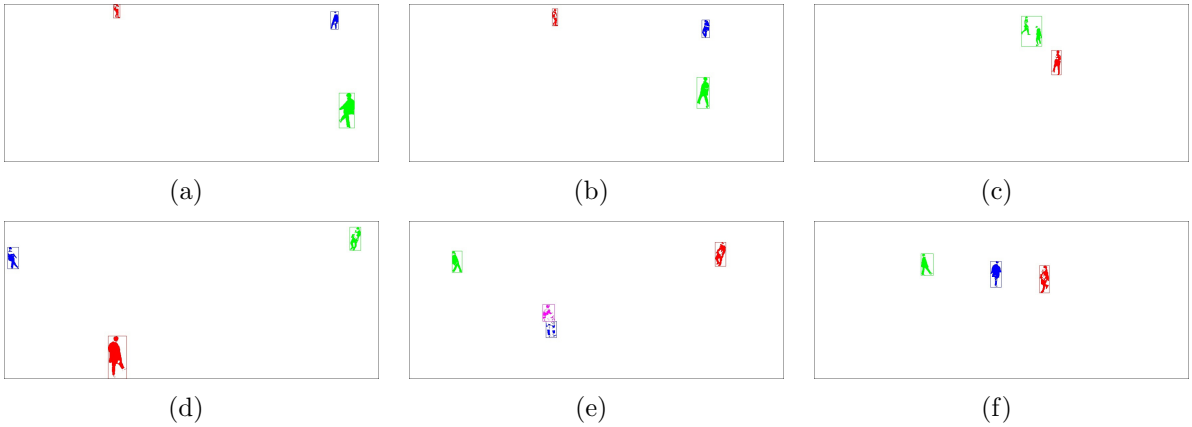


FIGURE 3. Foreground images segmentation

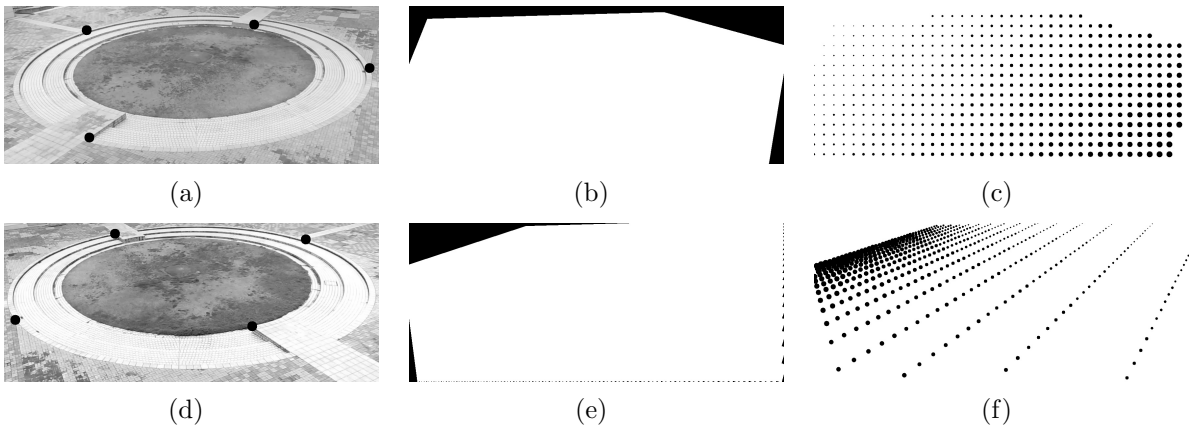


FIGURE 4. Points pairing based on perspective transform matrix

(d) in Figure 4. The white parts in (b) and (e) are the corresponding areas. It should be noticed that the pixels do not correspond one to one as the upper left pixels in (c) are sparsely distributed in the lower right part in (f), while the lower right pixels in (c) gather in the upper left part in (f).

A standing point-based foreground pairing method was proposed. Firstly, it segments the foreground and selects those not adjacent to the image edge as the complete target. Then, the middle pixels of the target's lower edges, which are in the effective area, are chosen and regarded as the standing points. After the coordinate transformation, the sum of every two-point pair distance is calculated, and the two-point pairs are considered a good match corresponding to the smallest distance, the same as the foreground segmentation. Figure 5 shows the result. Matched foreground segmentation is set with the same color, and the others are set with gray color as (a) and (d) show. Our method performs

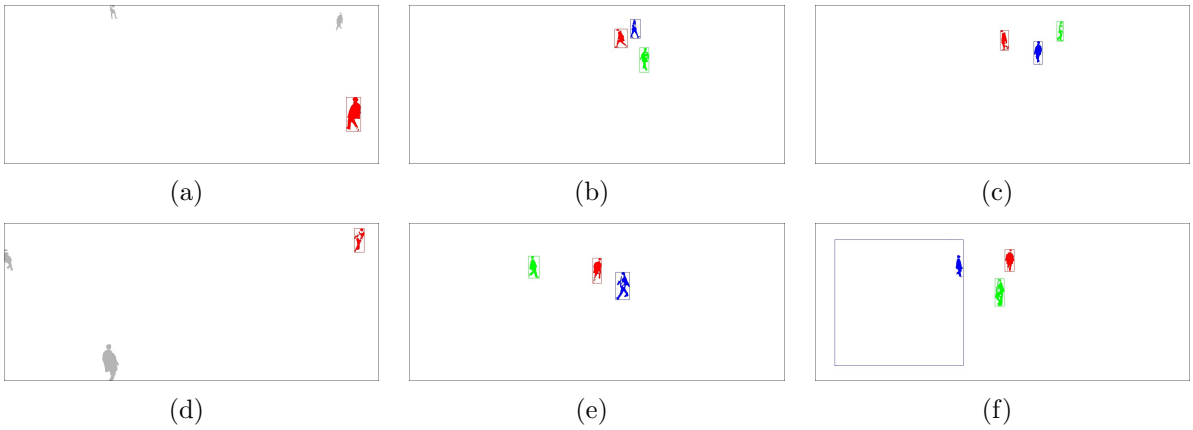


FIGURE 5. Foreground segmentation pairing

well with closed foreground segmentation like (b) and (e). Moreover, it is robust even when the standing point is far from the actual place, as (f) shows.

5. **Conclusion.** In this work, we propose a new foreground segmentation method based on the sum of within-cluster distance. Our process is simple: we ignore the between-cluster distance, which can deal with the limit that the number of clusters should be more than 2 in the silhouette coefficient, Calinski-Harabasz index, and Davies-Bouldin index. We also propose a foreground segmentation pairing algorithm based on perspective transformation. The result shows it is robust and performs well despite the inaccurate data acquired from the first step. We plan to apply this method to the pedestrian re-identification task in future work.

Acknowledgment. This work was supported by JST SPRING, Grant Number JPM JSP2136, and JSPS KAKENHI, Grant Number JP21K11964.

REFERENCES

[1] J. Dietlmeier, J. Antony, K. McGuinness and N. O'Connor, How important are faces for person re-identification?, *2020 25th International Conference on Pattern Recognition (ICPR)*, pp.6912-6919, 2021.

[2] S. Zhang, Y. Wang, T. Chai, A. Li and A. Jain, RealGait: Gait recognition for person re-identification, *arXiv Preprint*, arXiv: 2201.04806, 2022.

[3] G. Zhang, J. Liu, Y. Chen, Y. Zheng and H. Zhang, Multi-biometric unified network for cloth-changing person re-identification, *IEEE Transactions on Image Processing*, vol.32, pp.4555-4566, 2023.

[4] S. Sun and K. Inoue, Utilizing semantic information for color histogram-based person re-identification, *The 24th Meeting on Image Recognition and Understanding (MIRU)*, 2021.

[5] H. Han, M. Zhou, X. Shang, W. Cao and A. Abusorrah, KISS+ for rapid and accurate pedestrian re-identification, *IEEE Transactions on Intelligent Transportation Systems*, vol.22, pp.394-403, 2020.

[6] S. Sun and K. Inoue, A moving object detection method based on discrete Fourier transform, *ICIC Express Letters*, vol.16, no.11, pp.1235-1240, 2022.

[7] A. Ikotun, A. Ezugwu, L. Abualigah, B. Abuhaija and J. Heming, K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data, *Information Sciences*, vol.622, pp.178-210, 2023.

[8] B. Turkoglu, S. Uymaz and E. Kaya, Clustering analysis through artificial algae algorithm, *International Journal of Machine Learning and Cybernetics*, vol.13, pp.1179-1196, 2022.

[9] A. Bagirov, R. Aliguliyev and N. Sultanova, Finding compact and well-separated clusters: Clustering using silhouette coefficients, *Pattern Recognition*, vol.135, 109144, 2023.

[10] I. Ashari, E. Nugroho, R. Baraku, I. Yanda and R. Liwardana, Analysis of Elbow, Silhouette, Davies-Bouldin, Calinski-Harabasz, and Rand-Index evaluation on k-means algorithm for classifying flood-affected areas in Jakarta, *Journal of Applied Informatics and Computing*, vol.7, no.1, pp.95-103, 2023.

[11] F. Ros, R. Riad and S. Guillaume, PDBI: A partitioning Davies-Bouldin index for clustering evaluation, *Neurocomputing*, vol.528, pp.178-199, 2023.

[12] Y. Hou, L. Zheng and S. Gould, Multiview detection with feature perspective transformation, in *Computer Vision – ECCV 2020. Lecture Notes in Computer Science*, A. Vedaldi, H. Bischof, T. Brox and J. M. Frahm (eds.), Cham, Springer, 2020.

[13] J. Yu, N. Gao, Z. Meng and Z. Zhang, High-accuracy projector calibration method for fringe projection profilometry considering perspective transformation, *Optics Express*, vol.29, pp.15053-15066, 2021.