Paper

Shallow Layer Convolutional Features with Correlation Filters for UAV Object Tracking

Budi Syihabuddin, Suryo Adhi Wibowo, Agus D. Prasetyo, and Desti Madya Saputri

School of Electrical Engineering, Telkom University, Bandung, Indonesia

https://doi.org/10.26636/jtit.2022.150020

Abstract—In this paper, convolutional shallow features are proposed for unmanned aerial vehicle (UAV) tracking. These convolutional shallow features are generated by pre-trained convolutional neural networks (CNN) and are used to represent the target objects. Furthermore, to estimate the location of the target objects, an adaptive correlation filter based on the Fourier transform is used. This filter is multiplied with the convolutional shallow features by using pixel-wise multiplication in the Fourier domain. Then, the inverse of Fourier is performed to estimate the location of the target object, where its location is represented by the maximum value of the response map. Unfortunately, the target object always changes its appearance during tracking. Therefore, we proposed an updated model to address this issue. The proposed method is evaluated by using the UAV123_10fps benchmark dataset. Based on the comprehensive experimental results, the proposed method performs favorably against state-of-the-art tracking algorithms.

Keywords—CNN, convolutional features, correlation filter, object tracking, shallow layer, UAV tracking.

1. Introduction

Unmanned aerial vehicles (UAV) with remote sensing capabilities are used in many modern applications, such as object tracking and object recognition [1], [2]. In object tracking, numerous problems are encountered, such as aspect ratio change, background clutter, camera motion, fast motion, full occlusion, illumination variations, low resolution, out-of-view situations, partial occlusion, scale variation, presence of similar objects, and viewpoint change [3]. Those problems mean that object tracking systems have to comply with a number of requirements. Such systems must be capable of defining the next state, if they were given an initial state, for instance the initial object location or the initial object size. Object recognition can be useful for surveillance and human-computer interactions, but requires that numerous problems and issues be solved.

At the beginning of 2000, many researchers proposed a generative approach, i.e. suggested that an adaptive color histogram be used to identify objects [4]. The adaptive color histogram can be represented as an object, and a particle filter is used to estimate the next state. A similar method was used by [5] and [6]. The method is simple and easy, but that is why it suffers from a specific disadvantage. It is hard to identify an object using an adaptive color histogram if the distractor is characterized by similar color features. The particle filter uses Bayesian distribution to achieve a high level of accuracy. Distributions with many particles make the system more complex.

To compensate for the disadvantages of the previous method, the researchers proposed a discriminative approach based on boosting the classifier. This method uses a background model as initial information to come up with a robust object tracking algorithm. Grabner *et al*. proposed online learning relying on the Adaboost classifier for object tracking [7]. This approach was developed in paper [8] by proposing semi-supervised online boosting for object tracking and multiple instance learning based on boosting the classifier proposed in [9]. Zhang *et al*. added weight calculation based on distance and updated the model to approve accuracy of the system. Kalal *et al*. proposed a discriminative approach for long-term tracking based on tracking learning detection (TLD).

The discriminative approach using a boosting classifier suffers from certain disadvantages, i.e. a limited area taken as a positive sample to be represented as the target object. If the object moves quickly or abruptly, the method will have difficulty detecting it. This method can achieve good performance even if the object's color characteristics are similar to those of the distractor. However, by using the integral image, this method will run into a problem if occlusion is encountered.

To solve the occlusion problem, some papers recommended object representation based on a sparse coefficient vector. This method uses a generative approach and is particle filter to estimate the tracked object's location. A sparse coefficient vector has been researched by [12]–[14], and Wibowo *et al*. proposed a sparse coefficient vector to minimize computation time [15]. Computational time still remains a problem to be solved besides the fast motion and background clutter. As this method is being developed, it can be replaced by a correlation filter.

A correlation filter estimates the tracked object's location using the Fourier transform. Bolme *et al*. proposed the

minimum output sum of square error (MOSSE) approach that relies on a correlation filter and is capable of working adaptively [16]. This method can only work for simple linear classification problems. To boost the performance of the correlation filter, Henriques *et al.* used the ridge regression problem and the circulant matrix [17]. Other methods to increase the performance of the correlation filter include color histogram features, histogram of Gaussian (HOG) features and complementary learners [18]–[22]. For this paper, we proposed convolutional shallow features from a pre-trained CNN network to predict the movement of the object.

The remaining part of this paper is organized as follows. Section 2 discusses the correlation filter, and Section 3 presents the proposed method. Experimental results of the UAV123_10fps benchmark dataset and the result are presented and discussed in Section 4. Finally, Section 5 concludes this paper.

2. Correlation Filters

A correlation filter can be used to estimate the location of the targeted object. We work in the frequency domain to minimize computation time. Correlation filter h and the input that has been proceeded x (i.e. smoothen feature extraction) have to be transformed using the discrete Fourier transform (DFT). The output of h and x can be multiplied by each element to substitute the convolution process. Practically, DFT may be changed by means of the fast Fourier transform (FFT) to make the process more efficient. The maximum value of the inverse Fourier transform can be assumed as the location of the target object. Those processes can be described in the following manner:

$$\begin{aligned} x \otimes h &= F^{-1}(\hat{x} \odot \hat{h}^*) \,, \\ m &= F^{-1}(\hat{x} \odot \hat{h}^*) \,, \end{aligned} \tag{1}$$

where F^{-1} is the inverse Fourier transform, $\hat{x}$ is the smoothened feature extraction in the frequency domain, $\hat{h}$ is the correlation filter in the frequency domain, $*$ is complex conjugate, and $\odot$ is element-wise multiplication.

The correlation filter may use several data training approaches. In this case, it uses an image patch from the initial position from the first frame. To obtain the correlation filter, we can rely on minimization based on the following equation:

$$\min_{\bar{h}} \sum_i L\big(f(h,x_i), m_i\big) + \omega ||h||^2 , \tag{2}$$

where $L(\cdot)$ is the loss function.

The $L(f(h,x_i),m_i) = ||h \cdot x_i - m_i||^2$ can be expressed as:

$$\min_{\bar{h}} \sum_i ||h \cdot x_i - m_i||^2 + \omega ||h||_2^2 , \tag{3}$$

where h is the correlation filter, x is the smoothened feature extraction, m is the expected output, and ω is the control value to prevent overfitting.

Equation (3) can be simplified to:

$$h = (X^T X + \omega D)^{-1} X^T \mathbf{m} \,, \tag{4}$$

where D is the identity matrix, $\mathbf{m}$ is the labeled vector, and X is the training samples matrix. Since the computation takes place in the frequency domain, X^T has to be transformed into $X^H = (X^*)^T$, with $\mathbf{H}$ as the Hermitian matrix. To simplify the calculations for Eq. (4), we can use the circulant matrix as an approach. So, it can be expressed as:

$$\hat{h} = A \,\mathrm{diag}\left(\frac{\hat{x}}{\hat{x}^* \odot \hat{x} + \omega}\right) A^H \mathbf{m} \,, \tag{5}$$

with A being the DFT matrix. In the frequency domain, Eq. (5) will be:

$$\hat{h}_i = \frac{\hat{m}_i \odot \hat{x}_i^*}{\hat{x}_i \odot \hat{x}_i^* + \omega} \,. \tag{6}$$

3. Proposed Method

The CNN is a deep learning method that comprises convolutional layers, normalization layers and pooling layers. In the convolutional layers, it can be defined from the shallow layer to the deep layer. In this paper, we investigate the shallow layer from the convolutional layers to represent the target. We are using the CNN model proposed by Simonyan *et al.*, where the data set is trained by a big benchmark dataset [24]. We suggest checking [24] for architectural details and for the pre-trained CNN model. The proposed method is shown in Fig. 1.

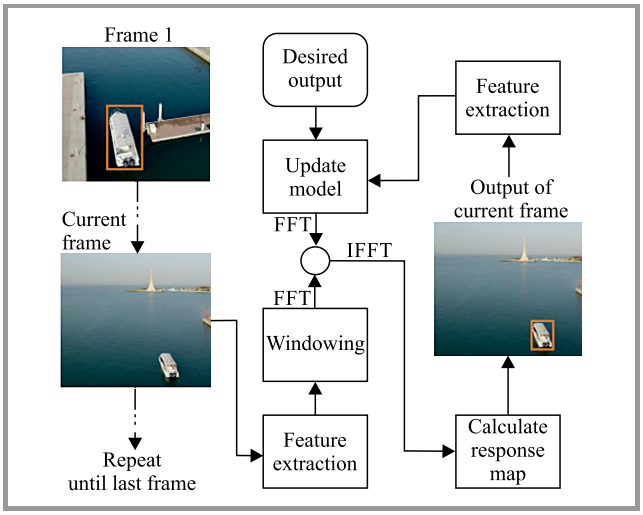


Fig. 1. Framework of the proposed method.

When we track an object from the same point of view but with the object moving, the appearance of the object will not be the same as it is in the initial state. It may be different in shape. To solve this problem, we must design a robust system for tracking objects. Updating the model is one of the potential methods. If we use the correlation filter, then

the filters will be updated in every frame. Referring to Eq. (6), for updating the correlation filters we use:

$$h_t = \frac{\beta_t}{\gamma_t + \omega} \ , \tag{7}$$

with h_t, β_t, γ_t, and ω being the correlation filters, numerator, denominator, and value at frame t, respectively. Equation (7) is solved using a linear system $I \times I$. For the numerator, we can use the following formula:

$$\beta_t = \alpha_1 \beta_{t-1} + \alpha_2 \left(m \odot x_t^*\right) \ , \tag{8}$$

where α_1, α_2, and β_{t-1} are weigh factor 1, weigh factor 2, and numerator for the previous frame $t-1$, respectively. The denominator can be solved by:

$$\gamma_t = \alpha_1 \gamma_{t-1} + \alpha_2 \sum_i x_t \odot x_t^* \ , \tag{9}$$

where x_t and γ_{t-1} are feature extraction at t and denominator at the previous frame $t-1$, respectively.

4. Experimental Results and Discussion

An updated model is needed to overcome changes in the appearance of the target object. In this step, variables ω, α_1, and α_2 exist, having the values of 0.001, 0.02, and 0.08, respectively. To validate the proposed method, referred to as csfUAVt, our tracker is evaluated with the use of the UAV123_10fps benchmark dataset. This dataset consists of 72 videos that contain several challenging problems, such as aspect ratio change, background clutter, camera motion, fast motion, full occlusion, illumination variation, low resolution, out-of-view, partial occlusion, scale variation, presence of a similar object, and viewpoint change. The proposed method is evaluated quantitatively using success plots based on the overlap ratio and precision plots based on the center location error. This evaluation is carried out following the one-pass-evaluation (OPE) protocol described in [3]. The proposed method was implemented using Matlab.

In this quantitative evaluation, the proposed method is compared with nine state-of-the-art tracking methods, such as ASLA [25], CSK [27], KCF [17], DSST [19], IVT [23], MOSSE [16], DCF, Struck [26], and TLD [11]. The results of the evaluation for the success plots of OPE are presented in Fig. 2. In the case of aspect ratio change, our proposed method obtains the best performance with a success rate of 0.289 and an overlap threshold value of 0.5. Meanwhile, TLD, Struck, DSST, and ASLA trackers are ranked second, third, fourth, and fifth with success rates of 0.287, 0.232, 0.227, and 0.212, respectively. For the TLD tracker, the features used are points and motion prediction is aided using optical flow. Meanwhile, for DSST, the feature used is the histogram of Gaussian (HOG). Based on the results for the aspect ratio change, our proposed method, using convolutional shallow features and correlation filters, offers the best performance compared to nine other tracker algorithms.

In the case of background clutter, the proposed method ranks second with a success rate of 0.279, while the winning Struck tracker outperforms the proposed method with a success rate of 0.361 for an overlap threshold value of 0.5. DSST, KCF, and DCF are ranked third, fourth, and fifth with success rates of 0.240, 0.232, and 0.231, respectively. Furthermore, the Struck tracker itself is developed based on a kernelized structured output support vector machine (SVM). Based on the results of these experiments, the tracker achieves superior performance compared to nine other trackers for background clutter problems.

In the case of camera motion, the proposed method provides the best performance, with a success rate of 0.376. Meanwhile, Struck, TLD, DSST, and MOSSE trackers are ranked second, third, fourth, and fifth with success rates of 0.280, 0.278, 0.244, and 0.237, respectively. The MOSSE tracker itself is a tracking algorithm based on adaptive correlation filters using the HOG feature. The experimental results show that the proposed method achieves better performance than nine other trackers for camera motion problems. Furthermore, for fast motion problems, the first, second, third, fourth, and fifth places are occupied by the proposed method, DSST, TLD, DCF, and CSK, respectively, with their respective success rates equaling 0.257, 0.160, 0.152, 0.149, and 0.136. In the case of fast motion, the proposed method shows better performance than the remaining tracking algorithms, with a success rate difference of 0.097 compared to the second rank.

Figure 2 shows the full occlusion problem in object tracking, with the entire shape of the target object being obscured by the distractor and making it invisible. In this case, the proposed method ranks first, with a success rate of 0.239, winning by a difference of 0.083 compared with the Struck tracker, ranking second. Meanwhile, ranks three, four, and five are occupied by TLD, MOSSE, and CSK, with their success rates equaling 0.149, 0.138, and 0.123, respectively. In the case of illumination variation, the proposed method ranks first with a success rate of 0.252, while the second, third, fourth, and fifth ranks are occupied by Struck, DSST, DCF, and KCF trackers achieving the success rates of 0.215, 0.187, 0.174, and 0.170, respectively.

In the case of low resolution, the Struck approach ranks first, with a success rate of 0.296, while the proposed method is placed fourth, rank with a success rate of 0.232. The second, third, and fifth ranks are occupied by TLD, ASLA, and MOSSE trackers, respectively. ASLA is a tracking algorithm that utilizes the sparse coefficient vector feature. Furthermore, in the case of out-of-view scenario, the proposed method ranks first, with a success rate of 0.5, outperforming the second-ranking DCF approach by 0.173. Meanwhile, DSST, KCF, and CSK occupy the third, fourth, and fifth place, with success rates of 0.327, 0.327, and 0.314, respectively.

Furthermore, in the case of partial occlusion, scale variation, similar objects, and viewpoint change, the proposed method offers superb results. It ranks first, showing the success rates of 0.375, 0.334, 0.438, and 0.342. The success plot of OPE is summarized in Table 1. Based on the

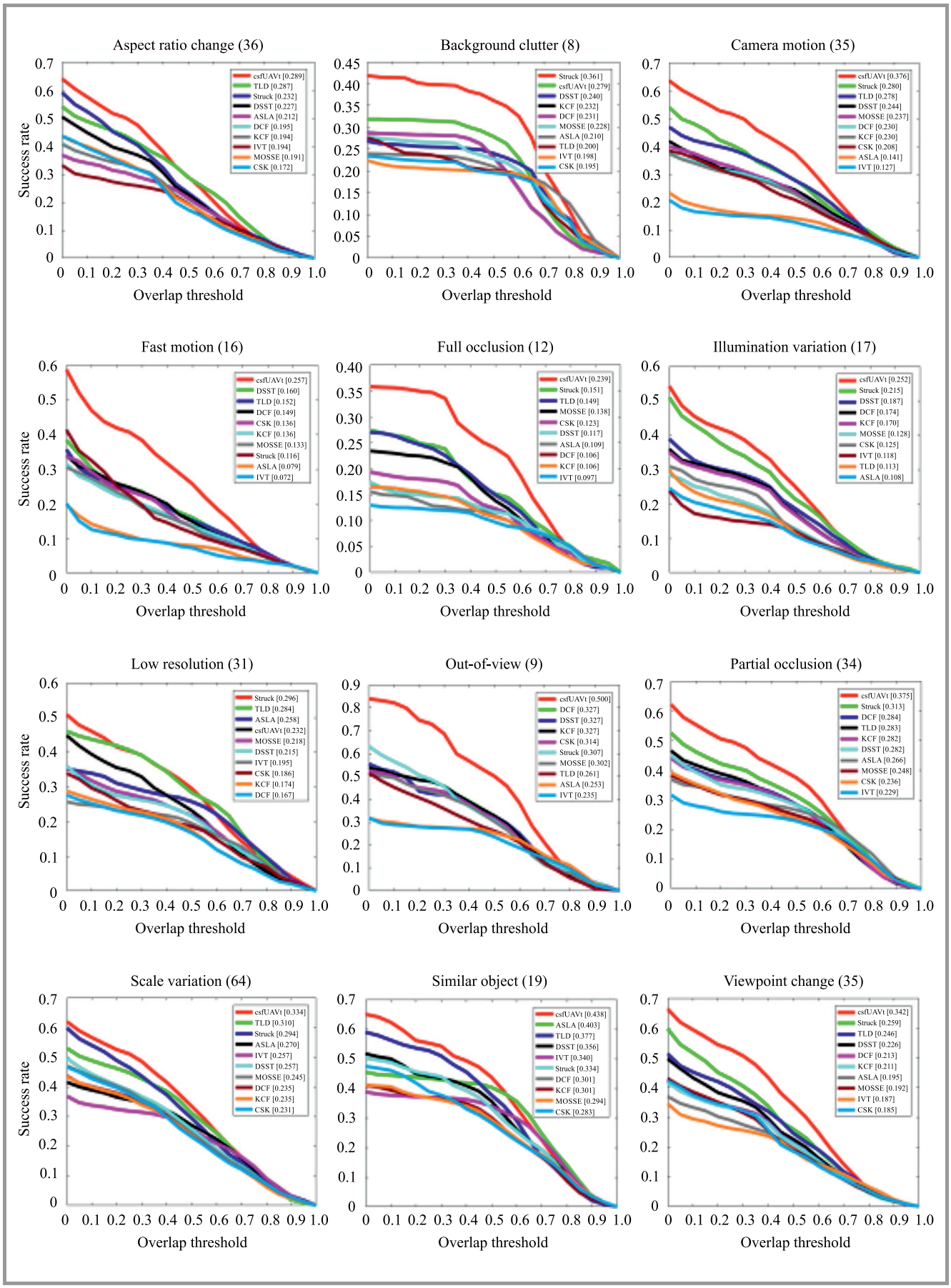


Fig. 2. Success plots of OPE for all types of problems encountered.

Table 1
Success plot of OPE

	Success plots of OPE for all types of problems encountered									
	csfUAVt	TLD	Struck	DSST	ASLA	DCF	KCF	IVT	MOSSE	CSK
Aspect ratio change	**0.289**	0.287	0.232	0.227	0.212	0.195	0.194	0.194	0.191	0.172
Background clutter	0.279	0.200	**0.361**	0.240	0.210	0.231	0.232	0.198	0.228	0.195
Camera motion	**0.376**	0.278	0.280	0.244	0.141	0.230	0.230	0.127	0.237	0.208
Fast motion	**0.257**	0.152	0.116	0.160	0.116	0.149	0.136	0.072	0.133	0.136
Full Occlusion	**0.239**	0.149	0.151	0.117	0.109	0.106	0.106	0.097	0.138	0.123
Illumination variation	**0.252**	0.113	0.215	0.187	0.108	0.174	0.170	0.118	0.128	0.125
Low resolution	0.232	0.284	**0.296**	0.215	0.258	0.167	0.174	0.195	0.218	0.186
Out-of-view	**0.500**	0.261	0.307	0.327	0.253	0.327	0.327	0.235	0.302	0.314
Partial occlusion	**0.375**	0.283	0.313	0.282	0.266	0.284	0.282	0.229	0.248	0.236
Scale variation	**0.334**	0.310	0.294	0.257	0.270	0.235	0.235	0.257	0.245	0.231
Similar object	**0.438**	0.377	0.334	0.356	0.403	0.301	0.301	0.340	0.294	0.283
Viewpoint change	**0.342**	0.246	0.259	0.226	0.195	0.213	0.211	0.187	0.192	0.185

experiment results, it is proved that the proposed method is more robust and useful than the nine other algorithms that are used as a benchmark for this specific UAV tracking application.

In addition the success rate, in this quantitative evaluation, the performance of our proposed method is also evaluated based on the precision plot parameters. The evaluation results for the precision plots of OPE are presented in Fig. 3. In the case of ratio change, the proposed method ranks first with a precision plot of 0.458 and a location error threshold of 20 pixels. Meanwhile, Struck, TLD, DSST, and DCF approaches are ranked second, third, fourth, and fifth, with success rates of 0.376, 0.376, 0.369, and 0.301, respectively. DCF is a tracking algorithm that relies on correlators and HOG as its features.

In the case of background clutter, the proposed method ranks second with a precision plot of 0.351. The first position, meanwhile, is occupied by the Struck method which outperforms the proposed approach thanks to its success rate of 0.443 and a location error threshold of 20 pixels. Meanwhile, DCF, KCF, and TLD are ranked third, fourth, and fifth, with prediction plots of 0.316, 0.316, and 0.294, respectively. Background clutter is a problem that is experienced in object tracking when background in close proximity to the target has the same color or texture as the target itself.

In the case of camera motion, the proposed method ranks first with a precision plot of 0.495. Meanwhile, Struck, TLD, DSST, and MOSSE are ranked second, third, fourth, and fifth with success rates of 0.379, 0.341, 0.326, and 0.295, respectively. In the case of camera motion, results of these experiments show that the proposed method offers better performance than 9 remaining trackers. Furthermore, in the case of fast motion, the second, third, fourth, and fifth ranks are occupied by the proposed method, DSST, CSK, TLD, and DCF, with their precision plot values amounting to 0.341, 0.257, 0.234, 0.197, and 0.175, respectively. In the case of fast motion, the proposed method shows better performance than other tracking algorithms, with its precision plot value differing by 0.084 compared to the second rank.

Figure 3 shows the full occlusion problem encountered in object tracking. In this case, the proposed method ranks first, with a precision plot of 0.435, winning by a margin of 0.098 compared with the second rank occupied by the Struck tracker. Meanwhile, third, fourth, and fifth ranks are occupied by MOSSE, TLD, and CSK approaches, with their precision plots equaling 0.3, 0.296, and 0.274, respectively. In the case of illumination variation, the proposed method ranks first with a success rate of 0.369, while second, third, fourth, and fifth ranks are occupied by Struck, DSST, DCF, and KCF methods, showing precision plot values of 0.325, 0.313, 0.236, and 0.227, respectively. Illumination variation is a problem encountered in object tracking, caused by significant changes in the illumination intensity in the region of the target object.

In the case of low resolution, the Struck approach ranks first with a precision plot of 0.5. Meanwhile, the proposed method occupies rank three, with a precision plot of 0.449. The second, third, and fifth ranks are occupied by TLD, DSST, and ASLA methods, with precision plot values of 0.455, 0.370, and 0.364, respectively. Low resolution is a problem encountered in object tracking due to the number of pixels in the ground-truth bounding box being smaller than 400 pixels. Furthermore, in the out-of-view scenario, the proposed method ranks first, with a precision plot of 0.640, outperforming the second-ranking a precision plot of 0.640, outperforming the second-ranking DSST approach by 0.221. Meanwhile, DCF, KCF, and Struck methods are ranked third, and fourth, respectively.

In the case of partial occlusion, scale variation, presence of a similar object, and viewpoint change, the proposed

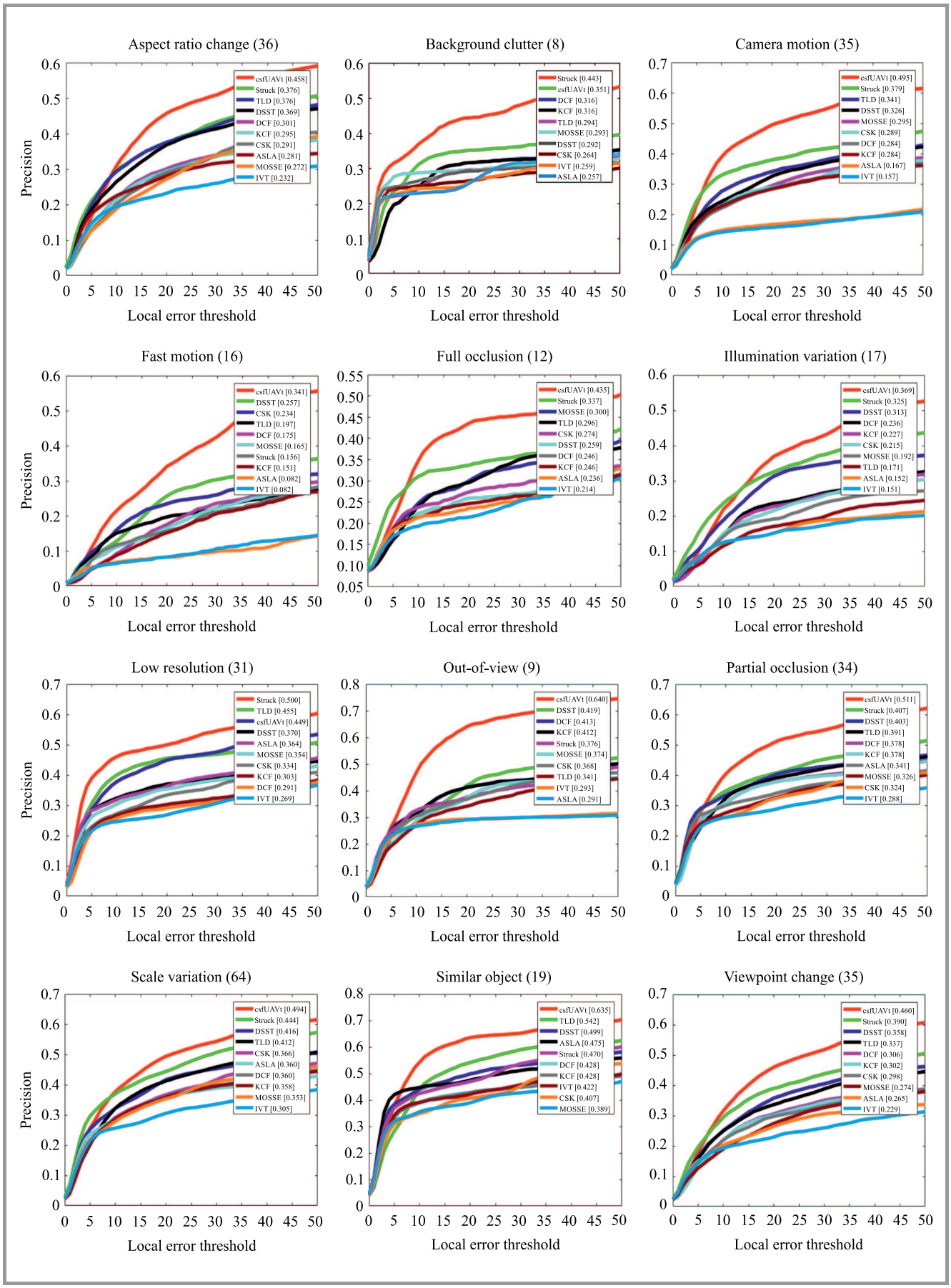


Fig. 3. Precision plots of OPE for all types of problems encountered.

Table 2
Precision plot of OPE

	Precision plots of OPE for all types of problems encountered									
	csfUAVt	TLD	Struck	DSST	ASLA	DCF	KCF	IVT	MOSSE	CSK
Aspect ratio change	**0.458**	0.376	0.376	0.369	0.281	0.301	0.295	0.232	0.272	0.291
Background clutter	**0.351**	0.294	0.443	0.292	0.257	0.316	0.316	0.259	0.293	0.264
Camera motion	**0.495**	0.341	0.379	0.326	0.167	0.284	0.284	0.157	0.295	0.289
Fast motion	**0.341**	0.197	0.156	0.257	0.082	0.175	0.151	0.082	0.165	0.234
Full occlusion	**0.435**	0.296	0.337	0.259	0.236	0.246	0.246	0.214	0.300	0.274
Illumination variation	**0.369**	0.171	0.325	0.313	0.152	0.236	0.227	0.151	0.192	0.215
Low resolution	0.449	0.455	**0.500**	0.370	0.364	0.291	0.303	0.269	0.354	0.334
Out-of-view	**0.640**	0.341	0.376	0.419	0.291	0.413	0.412	0.293	0.374	0.368
Partial occlusion	**0.511**	0.391	0.407	0.403	0.341	0.378	0.378	0.288	0.326	0.324
Scale variation	**0.494**	0.412	0.444	0.416	0.360	0.360	0.358	0.305	0.353	0.366
Similar object	**0.635**	0.542	0.470	0.499	0.475	0.428	0.428	0.422	0.389	0.407
Viewpoint change	**0.460**	0.337	0.390	0.358	0.265	0.306	0.302	0.229	0.274	0.298

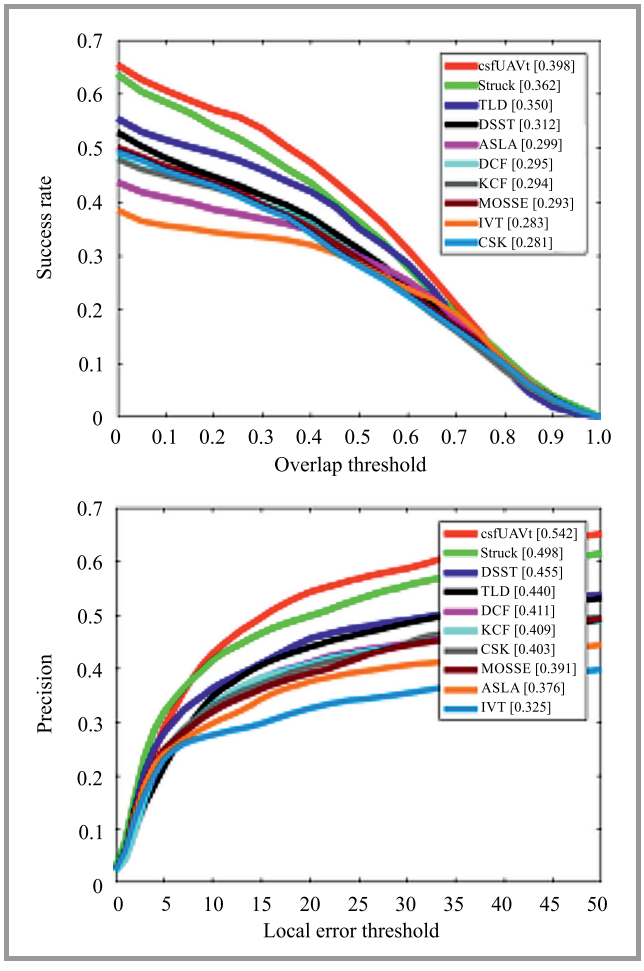


Fig. 4. Success plot and precision plot of OPE.

method offers superb results, ranking first in all the above-mentioned scenarios and exhibiting precision plot values of 0.511, 0.494, 0.635, and 0.460, respectively. The precision plot of OPE is summarized in Table 2. Based on the results of these experiments, it is proved that the proposed method is more precise than the nine algorithms that are used as a benchmark. Scale variation is a problem encountered in object tracking and influenced by the ratio between the bounding box in the first frame and in the latest out-of-range frame. Meanwhile, a similar object involves the presence of a distractor that has a similar color or texture to those of the target, and a viewpoint change is a problem caused by the difference in the target observation point, occurring between the first and the current frame.

After the success rate and precision plot have been calculated, each problem with UAV tracking is obtained. The average of each success rate and precision plot is calculated. The results of those calculations are represented in Fig. 4. In terms of the success rate of OPE, the proposed method ranks first, with a success rate of 0.398. Meanwhile, Struck, TLD, DSST, and ASLA approached occupy second, third, fourth, and fifth places, with their respective success rate values of 0.362, 0.350, 0.312, and 0.299.

In the case of precision plot, the proposed method also ranks first, with a precision plot value of 0.542, outperforming the second-ranking Struck methods by 0.044. Meanwhile, DSST occupies the third place, with a precision plot value of 0.455 and a location error threshold of 20 pixels. The fourth and fifth places are occupied by TLD and DCF, offering precision plot values of 0.440 and 0.411, respectively.

5. Conclusion

In this paper, convolutional features are taken from a CNN pre-trained on the shallow layer and harnessed using framework correlation filters. To solve the problem of changes in the appearance of the target object during tracking, the model is updated by correlation filters. In this updated model, numerator and denominator variables affecting the

correlation filters are worked out. To validate the proposed method, an experiment using the UAV123_10fps benchmark dataset was performed.

Based on the results of a quantitative evaluation relying on such parameters as success plots and precision plots, the proposed method ranks first in all scenarios, beating 9 other state-of-the-art tracking algorithms, with the average success plots of OPE equaling 0.398, and the average precision plots of OPE amounting to 0.542.

Acknowledgements

This study was by the Ministry of Research, Technology and Higher Education of the Republic of Indonesia under the *Penelitian Dasar Kompetitif Nasional* 2019–2021 project.

References

[1] X. Qin and T. Wang, "Visual-based tracking and control algorithm design for quadcopter UAV", in *Proc. of the Chinese Control and Decision Conf. CCDC 2019*, Nanchang, China, 2019 (DOI: 10.1109/CCDC.2019.8832545).

[2] Z. Zheng and H. Yao, "A method for UAV tracking target in obstacle environment", in *Proc. Chinese Autom. Congr. CAC 2019*, Nanchang, China, 2019, pp. 4639–4644 (DOI: 10.1109/CCDC.2019.8832545).

[3] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking", in in *Computer Vision – ECCV 2016. 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. *LNCS*, vol. 9905, pp. 445–461. Cham: Springer, 2016 (DOI: 10.1007/978-3-319-46448-0_27).

[4] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter", *Image and Vis. Comput.*, vol. 21, no. 1, pp. 99–110, 2003 (DOI: 10.1016/S0262-8856(02)00129-4).

[5] S.-K. Weng, C.-M. Kuo, and S.-K. Tu, "Video object tracking using adaptive Kalman filter", *J. of Visual Commun. and Image Represent.*, vol. 17, no. 6, pp. 1190–1208, 2006 (DOI: 10.1016/j.jvcir.2006.03.004).

[6] S. A. Wibowo, H. Lee, E. K. Kim, and S. Kim, "Tracking failures detection and correction for face tracking by detection approach based on fuzzy coding histogram and point representation", in *Proc. of the Int. Conf. on Fuzzy Theory and Its Appl. iFUZZY 2015*, Yilan, Taiwan, 2015, pp. 34–39 (DOI: 10.1109/iFUZZY.2015.7391890).

[7] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting", in *Proc. of the 17th British Machine Vision Conf. BMVC 06*, Edinburgh, Scotland, 2006 (DOI: 10.5244/C.20.6).

[8] H. Grabner, C. Leitsner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking", in Computer Vision – ECCV 2008 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I, D. Forsyth, P. Torr, and A. Zisserman, Eds. *LNCS*, vol. 5302, pp. 234–247. Berlin, Heidelberg: Springer, 2008 (DOI: 10.1007/978-3-540-88682-2_19).

[9] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning", *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, 2011 (DOI: 10.1109/TPAMI.2010.226).

[10] K. Zhang and H. Song, "Real-time visual tracking via online weighted multiple instance learning", *Pattern Recogn.*, vol. 46, no. 1, pp. 397–411, 2013 (DOI: 10.1016/j.patcog.2012.07.013).

[11] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection", *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, 2012 (DOI: 10.1109/TPAMI.2011.239).

[12] X. Mei, H. Ling, Y. Wu, E. P. Blasch, and L. Bai, "Efficient minimum error bounded particle resampling L1 tracker with occlusion detection", *IEEE Trans. on Image Process.*, vol. 22, no. 7, pp. 2661–2675, 2013 (DOI: 10.1109/TIP.2013.2255301).

[13] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach", in *Proc. of IEEE Conf. on Comp. Vision and Pattern Recogn.*, Providence, RI, USA, 2012, pp. 1830–1837 (DOI: 10.1109/CVPR.2012.6247881).

[14] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparse collaborative appearance model", *IEEE Trans. on Image Process.*, vol. 23, no. 5, pp. 2356–2368, 2014 (DOI: 10.1109/TIP.2014.2313227).

[15] S. A. Wibowo, H. Lee, E. K. Kim, and S. Kim, "Fast generative approach based on sparse representation for visual tracking", in *Proc. of the Joint 8th Int. Conf. on Soft Comput. and Intell. Syst. SCIS and 17th Int. Symp. on Adv. Intell. Sys. ISIS*, Sapporo, Japan, 2016, pp. 778–783 (DOI: 10.1109/SCIS-ISIS.2016.0169).

[16] D. S. Bolme, I. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filter", in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recogn. CVPR'10*, San Francisco, CA, USA, 2010, pp. 1401–1409 (DOI: 10.1109/CVPR.2010.5539960).

[17] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters", *IEEE Trans. of Pattern Anal. and Mach. Intell.*, vol. 37, no. 3, pp. 583–596, 2015 (DOI: 10.1109/TPAMI.2014.2345390).

[18] S. A. Wibowo, H. Lee, E. K. Kim, and S. Kim, "Multi-scale color features based on correlation filter for visual tracking", in *Proc. of the 1st Int. Conf. on Sig. and Sys. ICSigSys*, Bali, Indonesia, 2017, pp. 272–277 (DOI: 10.1109/ICSIGSYS.2017.7967055).

[19] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking", in *Proc. of the British Mach. Vis. Conf. BMVC'14*, Nottingham, UK, 2014 (DOI: 10.5244/C.28.65).

[20] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning", in *Computer Vision – ECCV 2014. 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. *LNCS*, vol. 8693, pp. 127–141. Cham: Springer, 2014 (DOI: 10.1007/978-3-319-10602-1_9).

[21] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: complementary learners for real-time tracking", in *Proc. of the IEEE Conf. on Comp. Vis. and Pattern Recogn. CVPR'16*, Las Vegas, NV, USA, 2016, pp. 1401–1409 (DOI: 10.1109/CVPR.2016.156).

[22] S. A. Wibowo, H. Lee, E. K. Kim, and S. Kim, "Visual tracking based on complementary learners with distractor handling", *Mathem. Probl. in Engin.*, vol. 2017, article ID 5295601, 2017 (DOI: 10.1155/2017/5295601).

[23] D. A. Ross, J. Lim, R. S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking", *Int. J. of Comp. Vision*, vol. 77, no. 1, pp. 125–141, 2008 (DOI: 10.1007/s11263-007-0075-7).

[24] K. Simonyan and A. Zisserman, "Very deep convolutional neural networks for large-scale image recognition", in *Proc. of the 3rd Int. Conf. on Learn. Represent.*, San Diego, CA, USA, 2015 [Online]. Available: https://arxiv.org/pdf/1409.1556.pdf

[25] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model", in *Proc. of the Int. Conf. on Comp. Vis. and Pattern Recogn.*, Providence, RI, USA, 2012 (DOI: 10.1109/CVPR.2012.6247880).

[26] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels", in *Proc. of the Int. Conf. on Comp. Vision*, Barcelona, Spain, 2011 (DOI: 10.1109/ICCV.2011.6126251).

[27] F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels", in *Computer Vision – ECCV 2012. 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012. Proceedings, Part IV*, A. Fitzgibbon *et al.*, Eds. *LNCS*, vol. 7575, pp. 702–715. Berlin, Heidelberg. Springer, 2012 (DOI: 10.1007/978-3-642-33765-9_50).

Budi Syihabuddin received his B.Sc. and M.Sc. in Telecommunication Engineering from the School of Electrical Engineering, Telkom University, Bandung, Indonesia, in 2008 and 2012, respectively. In 2010, he joined Telkom University as a lecturer in telecommunication engineering and a microwave laboratory researcher. His interests and publications are in RF microwave devices, wireless communication systems and embedded systems.

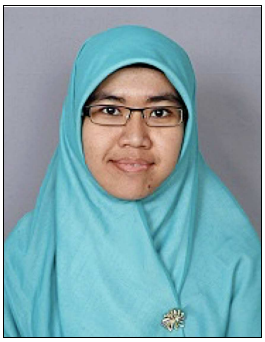
https://orcid.org/0000-0002-2322-2293
E-mail: budisyihab@telkomuniversity.ac.id
School of Electrical Engineering
Telkom University
Bandung, Indonesia, 40257

Suryo Adhi Wibowo received a B.Sc. degree from Telkom Institute of Technology, Indonesia, in 2009, an M.Sc. degree from Telkom Institute of Technology, Indonesia, in 2012, and a Ph.D. from the Department of Electrical and Computer Engineering, Pusan National University, Busan, Korea, in 2018. Now, he is a lecturer in telecommunication engineering and a researcher at the Image Processing & Vision (IMV) Laboratory. His research interests include intelligent systems, computer vision, computer graphics, virtual reality and machine learning.

https://orcid.org/0000-0002-3084-8534
E-mail: suryoadhiwibowo@telkomuniversity.ac.id
School of Electrical Engineering
Telkom University
Bandung, Indonesia, 40257

Agus D. Prasetyo received a B.Sc. degree in 2009 and an M.Sc. degree in 2013, both in Telecommunication Engineering from Telkom Institute of Technology, Bandung, Indonesia. He has been a lecturer in telecommunication engineering and a researcher at the satellite and radar laboratory, Telkom University, Bandung, Indonesia, since 2014. His interests and publications are in electromagnetic devices, antenna design, radar and satellite communication systems.

https://orcid.org/0000-0001-8880-3606
E-mail: adprasetyo@telkomuniversity.ac.id
School of Electrical Engineering
Telkom University
Bandung, Indonesia, 40257



Desti Madya Saputri received a B.Sc. degree in 2009 and an M.Sc. degree in 2012, in Telecommunication Engineering from the School of Electrical Engineering, Telkom Institute of Technology, Bandung, Indonesia. She has been a lecturer in telecommunication engineering and a researcher at the Communication Laboratory, Telkom University, Bandung, Indonesia, since 2012. Her interests and publications are in multiple access, coding theory and signal processing for wireless communications.

https://orcid.org/0000-0002-9200-9816
E-mail: destimadyasaputri@telkomuniversity.ac.id
School of Electrical Engineering
Telkom University
Bandung, Indonesia, 40257