

Face Recognition using Correlation Filters and Support Vector Machine in Machine Learning Approach

Hoang Long[†], Oh-Heum Kwon^{††}, Suk-Hwan Lee^{***}, Ki-Ryong Kwon^{****}

ABSTRACT

Face recognition has gained significant notice because of its application in many businesses: security, healthcare, and marketing. In this paper, we will present the recognition method using the combination of correlation filters (CF) and Support Vector Machine (SVM). Firstly, we evaluate the performance and compared four different correlation filters: minimum average correlation energy (MACE), maximum average correlation height (MACH), unconstrained minimum average correlation energy (UMACE), and optimal-tradeoff (OT). Secondly, we propose the machine learning approach by using the OT correlation filter for features extraction and SVM for classification. The numerical results on National Cheng Kung University (NCKU) and Pointing'04 face database show that the proposed method OT-SVM gets higher accuracy in face recognition compared to other machine learning methods. Our approach doesn't require graphics card to train the image. As a result, it could run well on a low hardware system like an embedded system.

Key words: correlation filters, MACE, MACH, UMACE, OT, SVM, machine learning, face recognition

1. INTRODUCTION

Correlation Filters (CF), for example, some of the most popular correlations fitters such as MACH, MACE, and their alternative[1,2], applied outstanding to various pattern recognition. In CF methods, we can construct an array of frequency domain for catching the desired parts and deemphasizing the undesired parts of training images to take changes in these images. For instance, face recognition is obtained by cross-correlating an input image with a filter and matching with the cor-

relation output. The images for testing will match the training images if we have a peak on the correlation plane. Fig. 1 presents the implementation of face recognition utilizing a CF and Fast Fourier Transforms (FFTs).

MACH filter is a type of correlation filter, which maximizes the correlation peak intensity like a response to the average training images. Another category of CF is MACE filters, which produce sharp correlations and offer excellent discrimination performance[3].

In this paper, we will develop a machine learning

* Corresponding Author: Ki-Ryong Kwon, Address: (48513) 45 Yongso-ro, Nam-gu, Busan, Pukyong National University, Daeyon campus, TEL: +82-51-629-6257, FAX: +82-51-629-6230, E-mail: kiryongkwon@gmail.com

Receipt date: Nov. 9, 2020, Revision date: Apr. 6, 2021
Approval date: Apr. 6, 2021

[†] Dept. of Artificial Intelligence Convergence, Pukyong National University
(E-mail: hoanglongdtvt2001@gmail.com)

^{††} Dept. of IT Convergence and Application Engineering, Pukyong National University
(E-mail: hheum@gmail.com)

^{***} Dept. of Computer Engineering, Dong-A University
(E-mail: skylee@dau.ac.kr)

^{****} Dept. of IT Convergence and Application Engineering, Pukyong National University

* This research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2020R1I1A306659411, 2020R1F1A1069124) and the Ministry of Trade, Industry, and Energy for its financial support of the project titled "the establishment of advanced marine industry open laboratory and development of realistic convergence content".

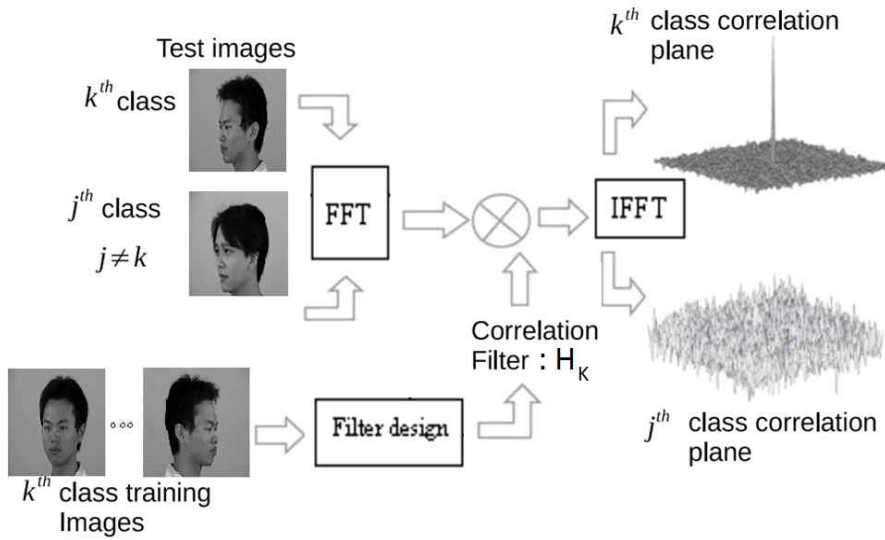


Fig. 1. Applying correlation filters in face recognition.

method, based on OT correlation filters in face recognition. The paper will summarize the related work in section 2, then present the MACE, MACH, UMACE, OT filter, and their combination with SVM in section 3. We show experimental results in section 4 and give a conclusion in section 5.

2. RELATED WORKS

We have two states: accept and reject in the facial security system, which are determined by the matching of a testing face with the face in the database. Correlation filters, such as the MACE filter, can handle this assignment completely. Marios Savvides [4] used a database of the facial expression from the lab of Advanced Multimedia Processing at CMU for performance evaluation of the MACE filter. Zhu et al. [5] introduced the feature correlation filter (FCF), an extended version of the MACE filter in feature space, tests on CMU's database. Another work on correlation filters is the MACH filter, which had been proposed by Nevel et al. [6], used for classifying the laser radar imagery. The MACH filter is also an efficient method to solve the problem in action recognition [7].

Random Forest (RF) based algorithm, which is an unsupervised machine learning method, is popular in computer vision and in solving facial recognition. The author in [8] presented an improved random forest-based method for face recognition by weighting the features when constructing tree predictors in the random forest. Several popular feature extraction methods have been used in [9,10] for face classification, such as Local Binary Patterns, Spatial Pyramid Based Local Descriptor and even Grey-scale intensity features.

In this paper, we first evaluate the facial recognition with four correlation filters: MACE, MACH, UMACE, and OT; then combine the OT with SVM to improve the recognition accuracy. We use the NCKU face database[11] and Pointing'04 face database[12] for testing because the CMU dataset is not free. Before moving to the next section, we review the ideal of SVM for solving classification problems. Fig. 2 shows two classification cases in SVM: linearly separable and non-linear separable. For the first case, SVM will find the decision boundary (hyperplane) with the largest margin to separate between two classes[13]. We can overcome the second case with the kernel trick, which has three popular types: linear, polynomial, and Radial

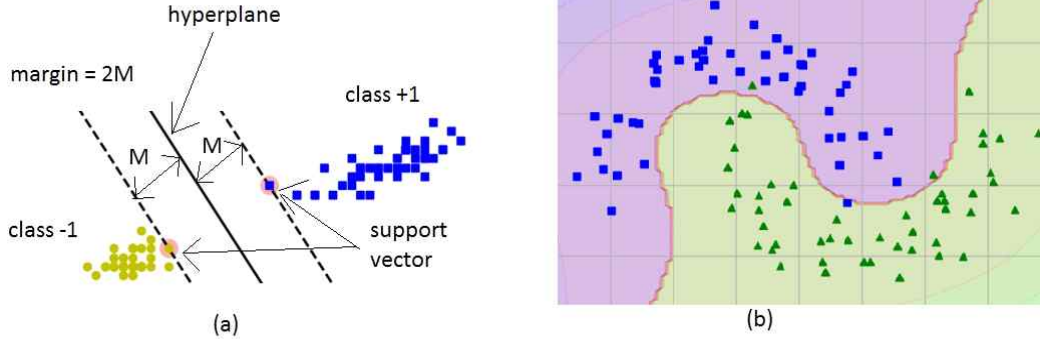


Fig. 2. SVM: (a) linearly separable and (b) non-linear separable.

Based Function – RBF. We use the RBF kernel for the proposed method because Yekkehkhany et al. indicate that the SVM with RBF gets higher classification results[14].

3. DESIGN OF CORRELATION FILTERS – SUPPORT VECTOR MACHINE

3.1. MACE, MACH, UMAC, and OT filter design

Assume that we have M training images, denoted by x_i ($i = 1, 2, \dots, M$), in which each image has the $d_1 \times d_2$ dimension. Vector h represents the correlation filter in a spatial domain.

We can convert the image from the spatial domain to the frequency domain by discrete Fourier transforms (DFT). X_i and H represent the 2D DFT of 2D images, x_i , and the 2D filter, h , respectively.

The correlation result of the i^{th} image and the filter is:

$$g_i(p, q) = h(p, q) \otimes x_i(p, q) \quad (1)$$

where \otimes represents the correlation operator; $x_i(p, q)$ represents the i^{th} image; and (p, q) are pixel values of the i^{th} image. We express the 2D DFT of the correlation function g_i by G . The MACE filter regulated the correlation plane's shape, $g_i(p, q)$. We solve the minimization problem of the average correlation energy (ACE) of the correlation plane from images of the training set to manage the shape of the correlation output[15].

The ACE can be expressed as:

$$E_{avg} = h^\dagger D h \quad (2)$$

Where diagonal matrix $D_i = X_i X_i^*$ is the power spectral density, and diagonal matrix $D = \frac{1}{M} \sum_{i=1}^M D_i$ is its mean from the training set.

The MACE filter is to minimize the ACE, so we find vector h that minimizes $h^\dagger D h$ with the condition $X^\dagger h = u$. The solution can be described as:

$$h = D^{-1} X (X^\dagger D^{-1} X)^{-1} u \quad (3)$$

We have already obtained the formula for a MACE filter. The other filters are also mentioned in this section.

The MACH filter minimizes the variance of a correlation plane between images in the training set and maximizes the output of the average correlation at the origin. These filters are observed to be statistically optimally for at least the Gaussian noise assumption. As a result, MACH filters are regularly more distortion tolerant than MACE filters. The MACH filter in the frequency area is defined as:

$$h = S^{-1} w \quad (4)$$

$$S = \frac{1}{M d_1 d_2} \sum_{i=1}^M (I_i - W) (I_i - W)^* \quad (5)$$

where, a $d_1 \times d_2$ diagonal matrix, S , compares the similarity of the images in training set with the class mean; w is a column vector of Fourier coefficients of the average training images; I_i and W are $d_1 \times d_2$ diagonal matrices which contain the

Fourier coefficients of i^{th} image and mean of the images along the diagonal in the training set, respectively. The UMACE filter, the special version of the MACH filter (substitute S by D), can be written as:

$$h = \mu \lambda D^{-1} w \quad (6)$$

where the term $\mu \lambda$ is constant, generally is set to 1, and does not affect the filter performance.

So far, we obtain the formula for the MACE, MACH, and UMACE filters. MACE is a particular instance of the well-known Synthetic Discriminant Function (SDF)[16]. The SDF filter has four inconsistent components, namely, output noise variance (ONV), average correlation height (ACH), average similarity measure (ASM), and average correlation energy (ACE). The ideal filter should reduce ONV, ACE, and ASM and increase ACH using optimal trade-off filters. The energy function is determined by equation 4[17].

$$E(h) = \alpha(ONV) + \beta(ACE) + \gamma(ASM) - \delta(ACH) \quad (7)$$

$$= \alpha h^+ C h + \beta h^+ D h + \gamma h^+ S h - \delta |h^+ m|$$

We minimize this function associated with one constraint, and the others are constant. The answer will be the optimal-tradeoff (OT) filter:

$$h = \frac{m}{\alpha C + \beta D + \gamma S} \quad (8)$$

where α , β , and γ are the nonnegative optimal trade-off parameters.

3.2 OT-SVM design

Fig. 3 and 4 show the proposed method. Suppose we have N different people in our dataset. We create each correlation filters h_k from M training images of the k^{th} person. We implement the Fourier transform for all training images of N classes and use the outcomes to design all the correlation filters H^k ($k=1, 2, \dots, N$), which are multiplied by the output of each training image after taking the Fourier transform. The inverse Fourier transformed was applied to the output of each multiplication, and give the final result, which use to determine the correlation peak in the traditional correlation filter methods. In our method, we use the final results to calculate the peak to side-lobe ratio (PSR) values, which are given by equation 7. The PSR values will be the input feature vectors of the SVM for the training phase. The process in the testing phase is similar to the training one, except that the SVM will predict the class for PSR values, also

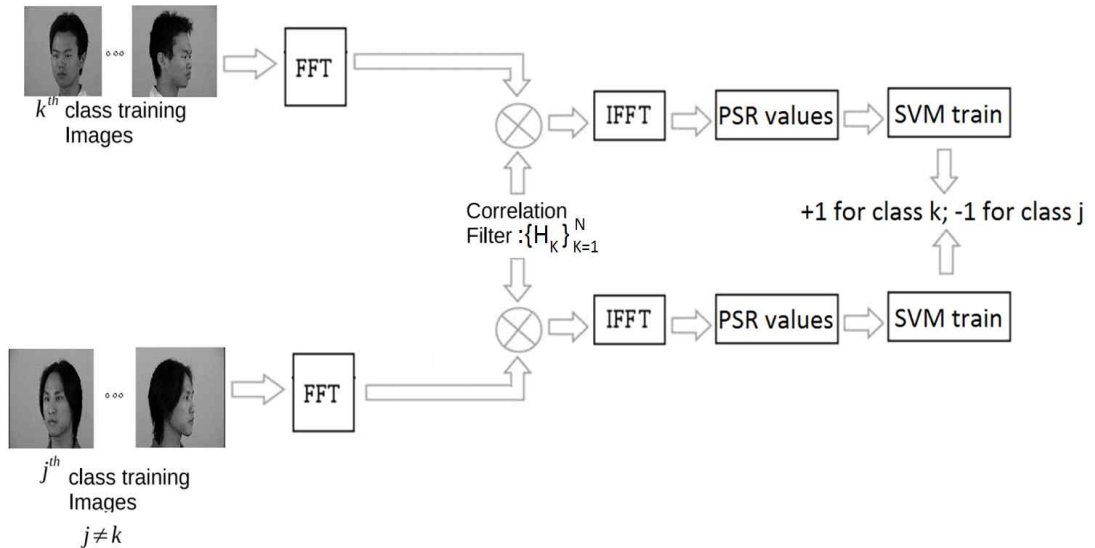


Fig. 3. The proposed method uses for the training set.

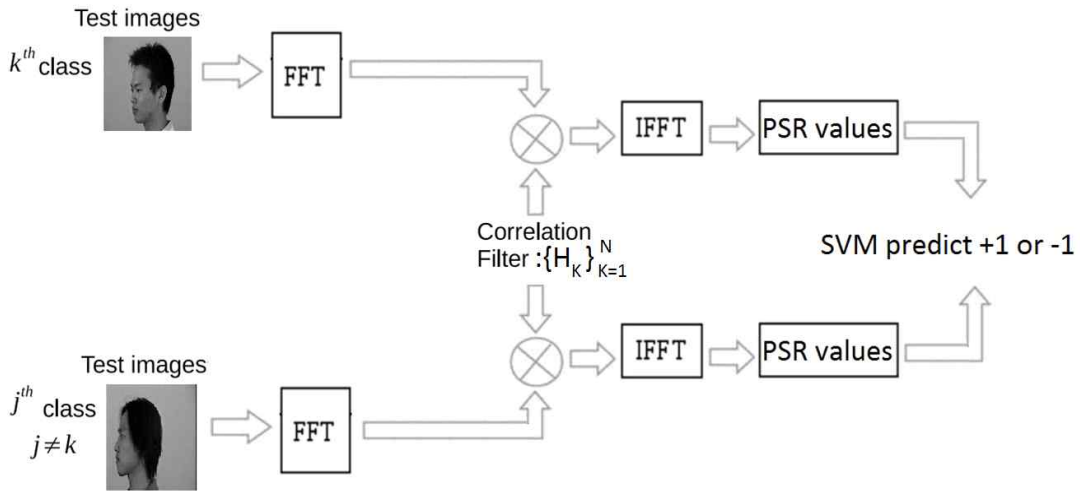


Fig. 4. The proposed method uses for the testing set.

the class of each testing image. Each of the testing images in the dataset will create the $1 \times N$ feature vector because we have N correlation filters.

$$PSR = \frac{\max(corr) - \text{mean}(corr)}{\text{std}(corr)} \quad (9)$$

where $\max(corr)$, $\text{mean}(corr)$, and $\text{std}(corr)$ are the maximum, mean, and the standard deviation of the correlation plane, respectively.

3. NUMERICAL RESULTS

The experiments run on an INTEL CPU i7 7700

with 32 GB RAM. First of all, we utilize the NCKU face database to estimate the performance of the correlation filters. We pick randomly 15 subjects from the database. There are 74 images in total for each subject, in which a half were generated by rotating the pair every 5 degrees from $+90^\circ$ (the right profile) to -90° (the left profile). Then, the authors in [10] use the commercial software in image processing for the horizontal flipping process on these existing 37 images to obtain the remaining 37 images. Fig. 5 is an example in the NCKU face database, pictures from 0° (frontal) to -90° (left

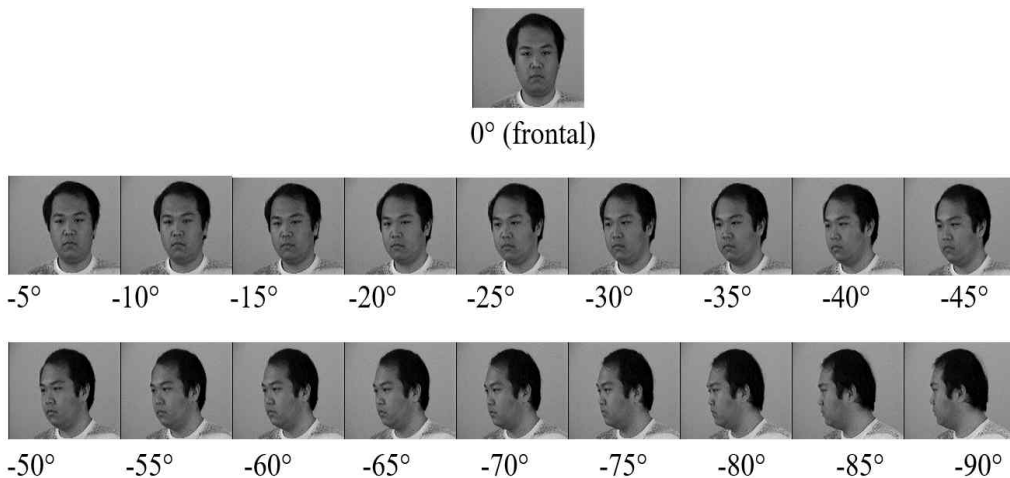


Fig. 5. The different views of one class in NCKU face database.

Table 1. The accuracy of various correlation filters on the NCKU database.

	<i>NCKU 15 Person Accuracy</i>	<i>NCKU 15 Person Precision</i>	<i>NCKU 15 Person Recall</i>	<i>NCKU 15 Person F1</i>
<i>MACE</i>	84%	0.8492	0.8929	0.8705
<i>MACH</i>	46.67%	0.5000	0.5644	0.5302
<i>UMACE</i>	82.22%	0.8222	0.8583	0.8399
<i>OT</i>	87.56%	0.8756	0.9048	0.8899
<i>MACE-SVM</i>	96.44%	0.9644	0.9684	0.9664
<i>MACH-SVM</i>	92.89%	0.9289	0.9350	0.9319
<i>UMACE-SVM</i>	96%	0.9600	0.9670	0.9635
<i>OT-SVM</i>	99.55%	0.9956	0.9958	0.9957

profile), obtained from this process.

We take a training/testing split on the data using 59 images for training and 15 for testing with each subject. The training data will be 885 images and 225 for testing data with the whole dataset. As seen in table 1, the MACE filter yields an accuracy of 84.00%, while MACH and UMACE are 46.67% and 82.22%, respectively. Fig. 6 shows the detailed result of the MACE filter. MACH filter gives the worst result. In contrast, the MACE filter classifies well for most of the classes, except for two groups. Each person, 5 and 7 in group 1, has an accuracy

of 46.7%. Three persons in group 2 have the same accuracy of 66.7%.

For each person in the dataset, for example, person A, we construct one correlation filter from 59 training images. For each testing image, we calculate the correlation output with the correlation filters. We have a peak on the correlation plane if the testing image is person A, and vice versa, as shown in Fig. 7.

So far, we have already compared the accuracy of four correlation filters: MACE, MACH, UMACE and OT. We evaluate the combination of the corre-

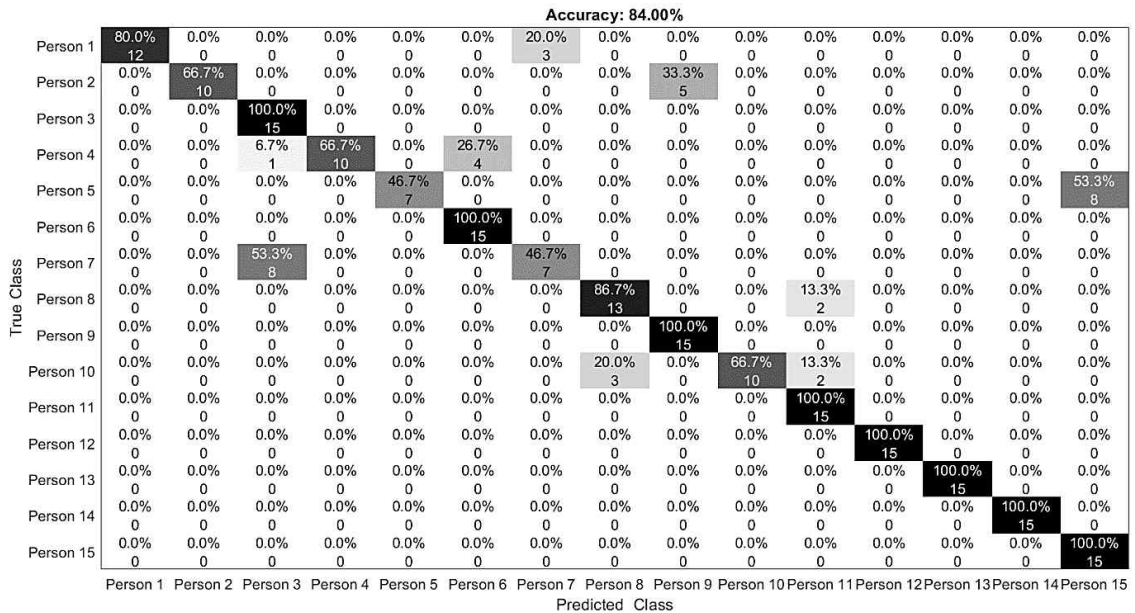


Fig. 6. The confusion matrix of MACE.

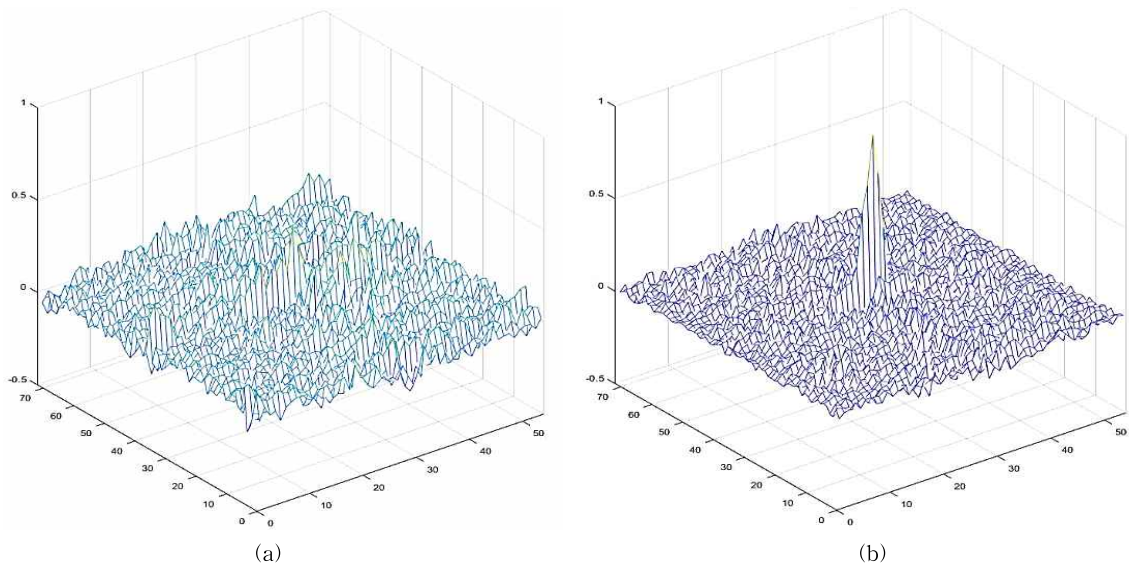


Fig. 7. The correlation peak shows how well the training and test images match: (a) No match and (b) Match.

lation filters and SVM in face recognition in the next step. We use the correlation filters to calculate the PSR values, which use as input vector of SVM.

Firstly, we compare the proposed method with the original correlation filter, as seen in table 1. The proposed method improves the accuracy of 12.44%,

46.22%, 13.78%, and 11.99% when compared to traditional filters MACE, MACH, UMAC, and OT, respectively (see table 1). All four filters get accuracy over 90% when using SVM to classify extracted features. Specifically, the OT-SVM got the highest accuracy of 99.55%. Fig. 8 gives the de-

Accuracy: 96.44%

	Person 1	Person 2	Person 3	Person 4	Person 5	Person 6	Person 7	Person 8	Person 9	Person 10	Person 11	Person 12	Person 13	Person 14	Person 15
Person 1	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 2	0.0% 0	86.7% 13	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	13.3% 2	0.0% 0	0.0% 0	0.0% 0
Person 3	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 4	0.0% 0	0.0% 0	6.7% 1	93.3% 14	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 5	0.0% 0	0.0% 0	0.0% 0	0.0% 0	86.7% 13	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	13.3% 2	0.0% 0	0.0% 0
Person 6	0.0% 0	0.0% 0	0.0% 0	13.3% 2	0.0% 0	86.7% 13	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 7	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 8	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 9	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 10	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 11	0.0% 0	0.0% 0	0.0% 0	6.7% 1	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	93.3% 14	0.0% 0	0.0% 0	0.0% 0	0.0% 0
Person 12	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0	0.0% 0
Person 13	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0	0.0% 0
Person 14	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15	0.0% 0
Person 15	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	0.0% 0	100.0% 15
	Person 1	Person 2	Person 3	Person 4	Person 5	Person 6	Person 7	Person 8	Person 9	Person 10	Person 11	Person 12	Person 13	Person 14	Person 15
	Predicted Class														

Fig. 8. The confusion matrix of MACE-SVM.

Table 2. The accuracy of various methods on the Pointing' 04 and NCKU database (N/A: Not Available).

<i>Method</i>		<i>Pointing'04 Accuracy</i>	<i>NCKU Accuracy</i>
<i>Machine Learning</i>	Weight-RF [8]	78.23%	80.42%
	WSRF (LBP) [9]	85.78%	85.89%
	WSRF (intensity) [9]	84.86%	85.56%
<i>Deep Learning</i>	AlexNet [17]	76.71%	N/A
	VGGNet [17]	69.29%	
	ResNet-50 [17]	89.00%	
	ConvNet (alpha) [18]	72.62%	N/A
	ConvNet (KL) [18]	90.00%	
	Model1 [19]	74.02%	N/A
	Model2 [19]	77.87%	
<i>Machine learning OT-SVM</i>		89.26%	86.29%

tailed result of the MACE-SVM in face recognition. The SVM-MACE has five classes under 100% accuracy. The SVM-MACH got the lowest accuracy at 92.89% but significantly improve with the original MACH (+46.22%).

Secondly, we use both NCKU and Pointing'04 face database to compare the proposed method with other machine learning methods. For a fair observation, we split the face database the same way that the authors mention in [8–9]. Specifically, 10% of Pointing'04 images are testing. With NCKU, we use a subset containing the first 37 images of all subjects (see [8]), and one-third of them are testing set. We chose $\gamma = 1$ and $\alpha = \beta = 0.5$ to create the optimal trade-off filter. The OT filter will extract the features from images, and SVM classifies these features. OT-SVM increases the accuracy by 11.03% on Pointing'04 and 5.87% on NCKU when compare to other machine learning methods (see table 2).

Table 2 shows the result of several deep learning approaches on the Pointing'04 dataset. The deep learning approach in Reference [18] evaluates the Pointing'04 dataset with three different deep learning architectures: AlexNet, VGGNet, and ResNet-50. Reference [19] applies a specific convolutional neural network structure: ConvNets, using two loss functions: alpha divergence (alpha) and Kullback-Leibler (KL). Two different CNN models are

evaluated on the Pointing'04 dataset (see Reference [20]). OT-SVM is more accurate than other deep learning methods, except ConvNet (KL) method. The ConvNet (KL) method requires a good GPU (an NVIDIA K40 GPU with 12GB memory), while a based-machine learning method such as OT-SVM runs well without the GPU.

5. CONCLUSION

Face recognition system is a computational technology that uses different algorithms that recognize and then validate the recorded faces with facial images stored in the database. Face recognition is an essential topic in computer vision, and many researchers have researched this subject in various ways concerning many applications such as surveillance systems. This paper presents an application of machine learning and correlation filters on face recognition. The OT filter obtains the features from images, and SVM classifies them. Experimental results show that the OT-SVM method gives a higher accuracy compared to the traditional filters and other machine learning and deep learning methods. OT-SVM doesn't require the GPU, and it could run well on limited hardware resources like embedded systems. For future work, we will integrate the proposed method with the embedded hardware and develop the full face rec-

ognition system.

REFERENCE

- [1] Q.W. Shahid and A.N. Alvi, "Object Tracking using MACH Filter and Optical Flow in Cluttered Scenes and Variable Lighting Conditions," *World Academy of Science, Engineering and Technology*, Vol. 60, pp. 709-712, 2009.
- [2] M. David and L.Y. Yu, "Face Recognition Subject to Variations in Facial Expression, Illumination and Pose using Correlation Filters," *Journal Computer Vision and Image Understanding*, Vol. 104, No. 1, pp. 1-15, 2006.
- [3] G. Verma and A. Sinha, "Design of Advanced Correlation Filters for Finger Knuckle Print Authentication Systems," *Proceedings of International Conference on Computer Vision and Image Processing*, pp. 47-56, 2017.
- [4] M. Savvides, B.V. Kumar, and P. Khosla, "Face Verification using Correlation Filters," *Proceedings of the 3rd IEEE Automatic Identification Advanced Technologies*, pp. 56-61, 2002.
- [5] X. Zhu, S. Liao, Z. Lei, R. Liu, and S.Z. Li, "Feature Correlation Filter for Face Recognition," *Proceedings of International Conference on Biometrics*, pp. 77-86, 2007.
- [6] A.J.V. Nevel and A. Mahalanobis, "Comparative Study of Maximum Average Correlation Height Filter Variants using Ladar Imagery," *Optical Engineering*, Vol. 42, No. 2, pp. 541-551, 2003.
- [7] J. Ahmed, S. Abbasi, and M.Z. Shaikh, "Fast Spatiotemporal MACH Filter for Action Recognition," *Machine vision and applications*, Vol. 24, No. 5, pp. 909-918, 2013.
- [8] R. Zhu, G. Sang, Y. Cai et al., "Head Pose Estimation with Improved Random Regression Forests," *Proceedings of the 8th Chinese Conference on Biometric Recognition (CCBR '13)*, pp. 457-465, 2013.
- [9] S. Gaoli, H. Chen, and Q. Zhao, "Head Pose Estimation with Improved Random Regression Forests," *Mathematical Problems in Engineering*, Vol. 2015, 2015.
- [10] K.T. Kim and J.Y. Choi, "Using Spatial Pyramid Based Local Descriptor for Face Recognition," *Journal of Korea Multimedia Society*, Vol. 20, No. 5, pp. 758-768, 2017.
- [11] The National Cheng Kung University Face Database, <http://www.datatang.com/data/14866> (accessed March 10, 2020).
- [12] Pointing'04 Database, <http://www-prima.inrialpes.fr/perso/Gourier/Faces/HPDatabase.html> (accessed March 10, 2020).
- [13] A. Géron, *Hands-On Machine Learning with Scikit-Learn & TensorFlow*, O'Reilly Media, Inc., CA 95472, USA, 2017.
- [14] B. Yekkehkhany, A. Safari, S. Homayouni, and M. Hasanlou, "A Comparison Study of Different Kernel Functions for Svm-Based Classification of Multi-Temporal Polarimetry SAR Data," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XL-2/W3, pp. 281-285, 2014.
- [15] A. Mahalanobis, B.V.K.V. Kumar, and D. Casasent, "Minimum Average Correlation Energy Filters," *Applied Optics*, Vol. 26, No. 17, pp. 3633-3640, 1987.
- [16] B.V.K.V. Kumar, J.A. Fernandez, A. Rodriguez, and V.N. Boddeti, "Recent Advances in Correlation Filter Theory and Application," *Proc. SPIE*, Vol. 9094, pp. 909404, 2014.
- [17] B.V.K.V. Kumar, D.W. Carlson, and A. Mahalanobis, "Optimal Trade-off Synthetic Discriminant Function Filters for Arbitrary Devices," *Optics Letters*, Vol. 19, pp. 1556-1558, 1994.
- [18] Y. Wang, W. Liang, J. Shen, Y. Jia, and L. F. Yu, "A Deep Coarse-to-fine Network for Head Pose Estimation from Synthetic Data," *Pattern Recognition*, Vol. 94, pp. 196-206, 2019.

- [19] B.B. Gao, C. Xing, C.W. Xie, J. Wu, and X. Geng, "Deep Label Distribution Learning with Label Ambiguity," *IEEE Transactions on Image Processing*, Vol. 26, No. 6, pp. 2825–2838, 2017.
- [20] S. Lee and T. Saitoh, "Head Pose Estimation Using Convolutional Neural Network," *Proceedings of the IT Convergence and Security*, pp. 164–171, 2017.



Hoang Long

He received his B.S. and M.S. degrees in Electronics Engineering at the University of Technology, Ho Chi Minh City, Vietnam in 2007 and 2011, respectively. Since September 2018, he's a Ph.D. student in the de-

partment of Artificial Intelligence Convergence at Pukyong National University, Korea. His research interests include Signal Processing and Machine Learning.



Oh-Heum Kwon

He received the BS degree in computer engineering from Seoul National University in 1988, and the MS and PhD degrees in computer science from the Korea Advanced Institute of Science and Technology in 1991 and 1996,

respectively. He is currently a professor in the Dept. of IT Convergence and Applications Engineering at the Pukyong National University, Pusan, Republic of Korea. His research interests include design and analysis of algorithms, medical image analysis, artificial intelligence, and machine learning.



Suk-Hwan Lee

He received a B.S., a M.S., and a Ph.D. degree in Electrical Engineering from Kyungpook National University, Korea in 1999, 2001, and 2004 respectively. He is currently a pro-

fessor in Department of Computer Engineering at Donga University. He is thesis editor-in-chief of KMMS journal. His research interests include multimedia security, digital image processing, and computer graphics.



Ki-Ryong Kwon

He received the B.S., M.S., and Ph.D. degrees in electronics engineering from Kyungpook National University in 1986, 1990, and 1994 respectively. He worked at Hyundai Motor Company from 1986–1988 and at

Pusan University of Foreign Language from 1996–2006. He is currently the Dean of Engineering College as well as a professor in Department of IT Convergence and Application Engineering at the Pukyong National University. He has researched University of Minnesota in USA in 2000–2002 with Post-Doc. He was the President of Korea Multimedia Society in 2015–2016. His research interests are in the area of digital image processing, multimedia security, bio-informatics, and machine learning.