Journal of Naval Sciences and Engineering
2020, Vol. 16, No.2, pp. 217-237
Electric-Electronic Engineering/Elektrik-Elektronik Mühendisliği

RESEARCH ARTICLE

**An ethical committee approval and/or legal/special permission has not been required within the scope of this study.*

CLUSTERING BASED FEATURE EXTRACTION METHODS FOR SEMAPHORE FLAG RECOGNITION*

Batuhan GÜNDOĞDU [1]
Deniz KUMLU [2]

[1] *National Defence University, Turkish Naval Academy, Department of Electrical and Electronics Engineering, Istanbul, Turkey, mbgundogdu@dho.edu.tr; ORCID: 0000-0002-9395-7519*

[2] *National Defence University, Turkish Naval Academy, Department of Electrical and Electronics Engineering, Istanbul, Turkey, dkumlu@dho.edu.tr; ORCID: 0000-0002-7192-7466*

Date Received: 07.10.2020 **Date of Acceptance: 19.11.2020**

ABSTRACT

Semaphore flag signaling is a visual communication system used by Naval vessels when the radio emissions are under strict control during electronic warfare. This paper presents an autonomous semaphore flag signaling recognition system that translates the RGB images into letters with a high performance at a low cost. A dataset is created for the semaphore flag signals for the English alphabet and the relative angles of the flags are acquired via morphological operations. The summarization of the flag locations is conducted with three methods: binary erosion the shrinking, k-means clustering and hierarchical agglomerative clustering. The resulting features with low dimensionality are then classified with support vector machine classifier with polynomial kernel. The cross-validation experiments show that the proposed methodology yields 99.76% accuracy, with no need for a Kinect sensor and computationally expensive neural network training that requires GPUs, as proposed by the similar works in the literature.

Keywords: *Semaphore Flag Recognition, Histogram of Oriented Gradients, Morphological Image Processing, Support Vector Machines.*

SİMAFOR BAYRAK TANIMA İÇİN KÜMELEME TABANLI ÖZNİTELİK ÇIKARMA YÖNTEMLERİ

ÖZ

Simafor ile bayrak muhaberesi elektronik harp esnasında yayın kontrolü yapılırken gemiler arasında çokça kullanılan bir görsel muhabere yöntemidir. Bu makalede RGB kamera imgeleri ile çalışan, yıüksek performanslı ve düşük maliyetli bir otomatik simafor tanıma sistemi önerilmektedir. Bu maksatla İngilizce simafor alfabesindeki harflere tekabül gelen bir vei seti oluşturulmuş ve morfolojik işlemler ile sancaklar arasındaki açılar otomatik olarak tespit edilmiştir. Sancak mevkilerinin özetlemesinde üç metot kullanılmıştır: ikili erozyon ile küçülme, k-ortalamalar ile öbekleme ve hiyerarşik toplayıcı öbekleme. Çok küçük boyutlu uzayda elde edilen öznitelikler ile karar destek makinaları kullanılarak sınıflandırma gerçekleştirilmiştir. Çapraz doğrulama deneyleri ile Kinct algılayıcıya ve hesap bakımından maliyetli sinir ağlarının çalışacağı GPU donanımlarına gereksinim duyulmaksızın %99.76'lık bir başarıma ulaşıldığı gözlemlenmiştir.

Keywords: *Simafor Sancak Tanıma, Yönlü Gradyenlerin Histogramı, Morfolojik İmge İşleme, Karar Destek Makinası.*

1. INTRODUCTION

Autonomous human-machine interface (HMI) systems have recently gained interest in the machine learning literature. This paper proposes an autonomous HMI system to facilitate semaphore flag signaling (SFS) recognition. SFS is a visual communication technique that was used for far field communications before the invention of telegraph. It is now mainly used by the Navy under emission control operations either during electronic warfare or while underway replenishment. It is usually employed by means of flags (or lights) and moving them to different angles to depict different letters. This set of symbols can be seen in Figure 1. The aim of this study is to translate the images of flag operators holding flags, into the corresponding letters. This work can be considered as an example of human posture and body gesture recognition applications.

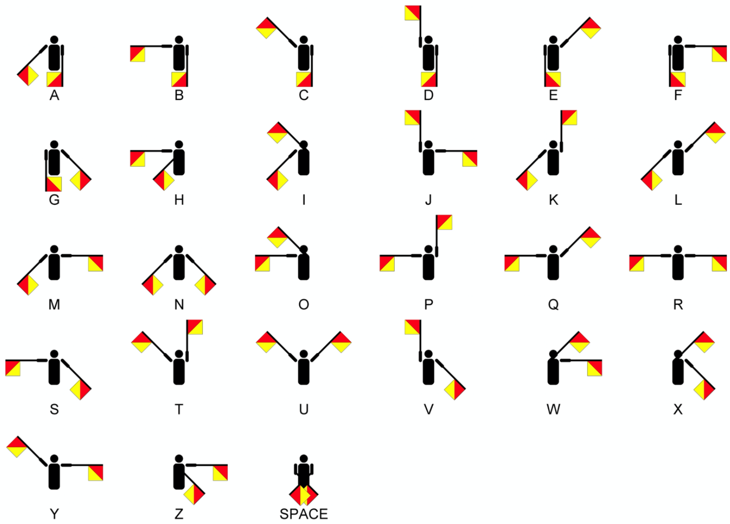

Figure 1. The SFS Alphabet.

Several studies have been conducted over the past decade on automated recognition of semaphore flag signaling. The initial work used Kinect sensors to address the problem, since in SFS the information is conveyed via the angles between arms and the body. In (Iwane, 2012), a rule-based table look-up methodology was proposed, acting on angles of head and elbow measured by a Kinect sensor. This work was implemented over a set of 14 Japanese semaphore characters. In (Hsieh & Shih, 2015), SFS was addressed to use robots as tangible learning companions. This work also depends on Kinect sensor measurements of human holding flags in order to recognize and implement robot interaction. The study was limited to run over 10 semaphore signals selected from the semaphore alphabet given in Figure 1.

To our knowledge, the first work dealing with the whole alphabet set was (Rachmad & Fuad, 2015), in which the geometric calculations were conducted on skeleton image points, obtained by a Kinect sensor. The work used three representative points out of Kinect measurements, i.e. the center of shoulder, right and left elbows and proposed calculating geometric attributes for each letter. Similarly, (Hung, Hsu & Chen, 2015) studied utilizing robots to enhance learning SFS, in the task called situated learning. The authors used 10 selected symbols of the semaphore flag alphabet using a Kinect sensor.

More recent work such as (Zhao et. al, 2016), proposed working on monocular camera images of semaphore flag signals, as Kinect sensor is rarely available on practical applications. They proposed using convolutional neural networks (CNN) trained on raw image pictures, on a selected set of 5 semaphore flag signals. Similarly, (Tian et. al, 2018) proposed using the state-of-the-art feature extractor called Openpose (Cao et. al, 2018), based on deep neural networks. They proposed recognizing the whole set of semaphore alphabet flags, detecting successfully the combinations of 7 main directional angles. More recently, (Motty, Yogitha, & Nandakumar, 2019) used pre-trained CNN's to implement an end-to-end semaphore flag recognition system that operates on images captured by camera. The most recent work approaches the problem with cameras and RGB images, eliminating the necessity for Kinect sensors, though it should be noted that such deep models require greater computational resources like GPUs to be able to provide real-time recognition. A summary of the related work in SFS recognition is also given in Table I.

Our main ambition with this study is to propose a robust feature extraction methodology for a low cost SFS classifier that works over the whole semaphore alphabet. For this, we introduce an area of interest detection methodology that makes use of morphological operations. We detect red regions that are expected to correspond to the semaphore flags and extract the location information of these regions by three methodologies: shrinking by binary erosion, k-means clustering and hierarchical agglomerative clustering (HAC). We compare these three methodologies with each other, as well as the performance of histogram of oriented gradients (HOG) features. As the Kinect sensor is rarely available in practical applications, the more recent research focuses on camera images. Nonetheless, the high-performance systems that use deep features such as Openpose or CNN bottlenecks require costly hardware (e.g. GPUs) for low latency operation, as reported by (Motty., Yogitha, & Nandakumar, 2019). The proposed technique in this work, on the other hand, works on RGB images, with high accuracy and at a comparatively low computational cost. We compare the performance of our system with the baselines with respect to both the accuracy and the run time metrics. In addition to the methodological and applicational novelties stated above, we provide the semaphore dataset RGB images and the feature vectors that we created available online, for further research.

2. METHODOLOGY

The technique proposed in this paper follows a series of image segmentation and morphological processing phases in order to obtain a low dimensional discriminative feature for classification.

We use RGB images as inputs and seek a low-cost subspace for model training, in order to alleviate the need of high performance GPUs, as necessitated by deep models like CNNs. We propose three such methods that seek to find the flag locations and compare them with baseline feature extractors both on accuracy and computation time metrics. Once the feature vectors are obtained, we train SVM-based classifiers to obtain the corresponding letters from the images.

Table 1. Summary of different works pertaining to semaphore flag recogniton.

Study	**Sensor**	**#Flag Signs**	**Classifier**	**Feature Representation**
(Iwane, 2012)	Kinect	14	Table look-up and model comparison	Angles of head and elbows
(Hsieh & Shih, 2015)	Kinect	10	Table look-up and model comparison	Skeleton images
(Rachmad & Fuad, 2015)	Kinect	26(full)	Table look-up and model comparison	Three points from skeleton images: Center point of shoulder, right and left wrist
(Hung, Hsu & Chen, 2015)	Kinect	10	Hand position detection for situated learning	Skeleton images
(Zhao et. al, 2016)	Camera	5	CNN	Raw RGB
(Tian et. al, 2018)	Camera	All 26 flags from 7 positional vectors	DNN(openpose)	Raw RGB
(Motty., et. al, 2019)	Camera	26(full)	Pretrained CNN	Raw RGB
This work	Camera	26(full)	SVM	Very low-D position indices HOG PCA

The proposed methodology of detecting the flags is as follows: The big chunks of red pixels that exist on the two flags are emphasized by differencing the red channel from the gray-scale mapping of RGB image.

These emphasized red regions are generally discontinuous due to the shadows on the flag, or due to the occlusion caused by the folding of the flag. We use a big enough median filter to address such discontinuities on the bright regions that correspond to the red portion of the semaphore flag. The resulting image is then quantized to be binary, where the red regions are white and the non-informative rest of the image is cast to black. The shape and size of the median filter should be chosen relative to the size of the flag. With the binary image involving the chunks of white regions that correspond to the flag positions, only the white pixel locations are taken into account and the integer valued pixel addresses are clustered to 2 centroid locations. In the end, all we wish to obtain is a feature vector for each image that would have the coordinates of the center of the two flags, because the information about the letters are conveyed with respect to the relative locations and the angle between the two flags. The feature vector can be expressed as:

$$\mathbf{v} = [R_x, R_y, L_x, L_y] \tag{1}$$

Where R and L stand for right and left, and the superscripts are the 2-D coordinates (See Figure 2).

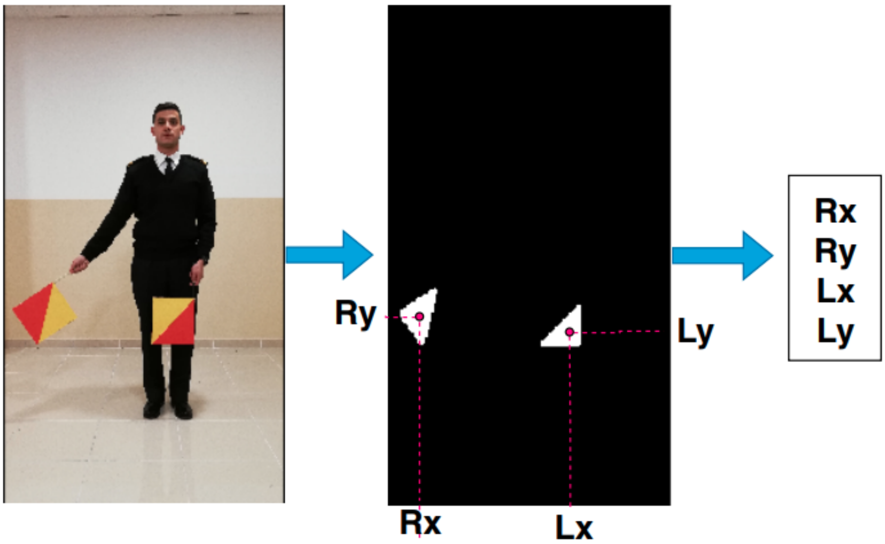

Figure 2. Feature extraction for red area detection.

We follow three schemes to obtain the centroid locations for the two clusters of white pixels:

- Shrinking by binary erosion (Rao, Prasada & Sarma, 1976).
- K-means clustering (Llody, 1982).
- Hierarchical agglomerative clustering (Day & Edelsbrunner, 1984).

2.1. Clustering by Shrinking

The shrinking algorithm conducts erosion on connected white pixels starting from the ones that touch the black pixels, i.e. the outer borders, until they are reduced to one pixel. The advantageous aspect of this algorithm is that it is guaranteed to summarize the right-hand image in Figure 2 as long as the white area is connected. However, the algorithm suffers from inconsistent run-time duration that is dependent on the size of the connected white area. With the shrinking algorithm, an object without holes erodes to a single pixel at or near its center of mass; on the other hand, an object with holes erodes to a connected ring lying midway between each hole and its nearest outer boundary. Such an effect can be seen on the sample in Figure 3. It should be noted that we apply a median filter that is applied prior to the morphological processing and the effect of this filter is to alleviate such problems.

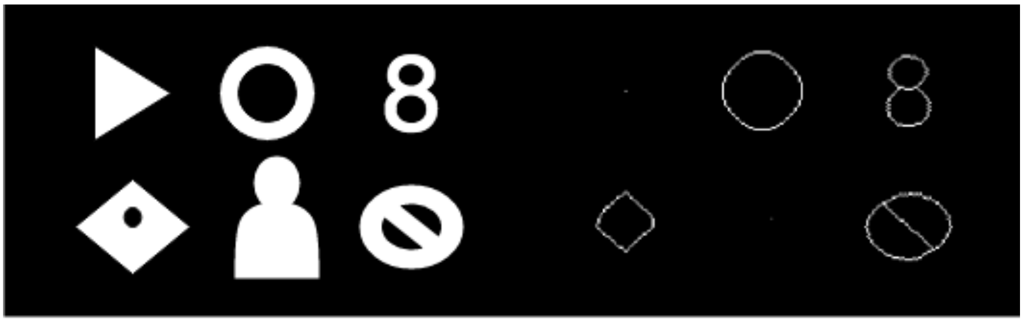

Figure 3. Shrinking behaviors of connected and non-connected objects, the original shapes are given on the left and the shrunk objects are on the right figure.

2.2. K-means Clustering

As the second algorithm, we used the k-means clustering to obtain the white pixel clusters, i.e. the flag centroids, since the shrinking algorithm may suffer from non-connected areas. A situation where the white pixels are not connected is quite likely due to the folding of the flags or the occlusion effects. The k-means clustering algorithm is appealing in this application since the number of clusters is definitely known to be two, since we have two flags. For this, the squared Euclidean distance between the addresses of the white pixels and the corresponding centroids are minimized. The assignments of the pixels to the left (L) and right (R) flags and the updates of the centroids are done via expectation minimization. The cost function in the maximization process can be stated as thus:

$$R_x, R_y, L_x, L_y = \underset{R_x, R_y, L_x, L_y}{\arg\min} \; J(R_x, R_y, L_x, L_y) \tag{2}$$

$$J = \sum_{(i,j)\in L} (i - L_x)^2 + (j - L_y)^2$$
$$+ \sum_{(m,n)\in R} (m - R_x)^2 + (n - R_y)^2 \quad \textbf{(3)}$$

K-means clustering approach is faster and more robust than the shrinking-based approach; however, it suffers from the initialization of the centroids. The resulting 4-dimensional vector may not be consistent over all set of images. Therefore, in order to address this aspect, we used the hierarchical agglomerative clustering (HAC) as the third clustering algorithm as a compromise between the shrinking-based approach and the k-means based approach.

2.3. Hierarchical Agglomerative Clustering

In the hierarchical agglomerative clustering, new clusters are formed at each iteration starting from the closest pixels. Euclidean distance metric-based dissimilarities are calculated between addresses of each white pixel pair, and then a new cluster is formed to involve the white pixel pair that has the shortest distance at each iteration. On the consequent iterations, white pixels are added to the group to which they are the closest. This procedure is continued until there are two groups left. At each iteration, the distance (D) between two clusters $\mathcal{A}$ and $\mathcal{B}$ is decided by the minimum distance between points that belong to them:

$$D(A,B) = \min_{a\in A, b\in B} d(a,b) \quad \textbf{(4)}$$

and the distance update formula between the fusion of the two clusters and another cluster is conducted likewise:

$$D(A \cup B, C) = \min(D(A,C), D(B,C)) \quad \textbf{(5)}$$

Where *d(a,b)* is the Euclidean distance metric between the locations of white pixels denoted by *a* and *b* (Müllner, 2011).

HAC algorithm could be stopped at any iteration and could potentially yield the best clusters so far. For this specific application we continue until there are two clusters, corresponding to the right and left flags.

2.4. Baseline Models

The low dimensionality representation denoted by the pixel locations is compared to two baseline methods. In the first methodology the principle component analysis (PCA) over the grayscale images is conducted to obtain the lowest dimensionality subspace representation that would give the maximum variance.

As the second baseline, we obtained the histogram of oriented gradients (HOG) (Dalal & Triggs, 2005) representation. HOG has never been used for semaphore recognition tasks, yet it can be seen as an appropriate feature set since it is generally used as a spatio-temporal descriptor that captures the angles in the images. Initially proposed for pedestrian detection, HOG has been successfully used in similar tasks like human action recognition (Ikizler & Duygulu, 2009) hand gesture recognition (Freeman & Roth, 1995), human detection (Zhu, Yeh, Cheng & Avidan, 2006) and explosive detection (Temlioglu, Erer & Kumlu, 2017). In HOG feature extraction, we divide an image of size $I_x \times I_y$ into blocks of $B_x \times B_y$ and calculate O orientated gradients within each block. The normalization of these gradients is taken within $C_x \times C_y$ cells and the histogram vectors of the gradients are calculated for each cell, yielding a feature vector of size:

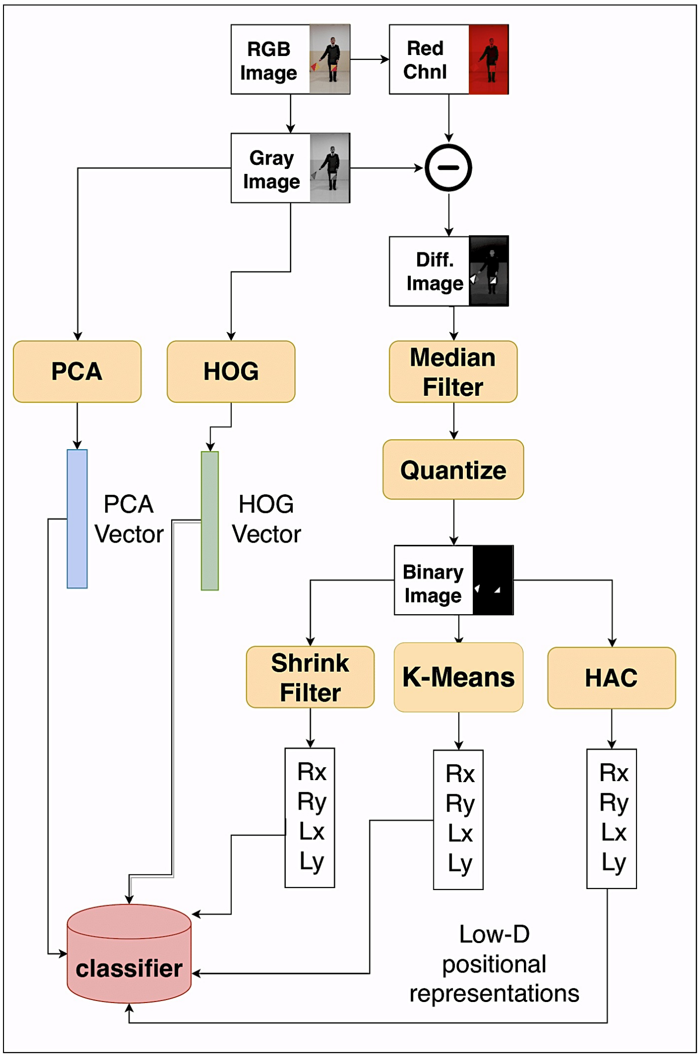

Figure 4. The flowchart of the feature extraction methods proposed in this paper.

$$HOG = S_{xS_y} B_x B_y O$$

$$S_x = \left(\frac{I_x}{C_x} - (B_x - 1) \right) \qquad (\mathbf{6})$$

$$S_y = \left(\frac{I_y}{C_y} - (B_y - 1) \right)$$

This high dimensional feature vector is expected to convey more discriminative information than the raw image and a PCA is also applied on this feature.

The feature vectors obtained by each of the methodologies are used to train a support vector machine-based classifier (Suykens & Vandewalle, 1999) in order to assess the discrimination power of the proposed models. The flowchart of the methodologies explained in this chapter can be seen in Figure 4.

3. EXPERIMENTS

For the experiments we created a semaphore flag signaling dataset over the whole alphabet and the proposed techniques were compared with various feature dimensions using both accuracy and computation time metrics.

3.1. Dataset

The semaphore dataset was created by the Naval Academy midshipmen and the semaphore flags provided for educational purposes. Four different students took part in the preparation of the dataset, who has different body sizes. In addition to using different height subjects in dataset creation, we also allowed same lighting and camera distance changes to provide a naturalistic variation of images. The source code of the methodologies explained in this paper and the dataset can be found online at (Gundogdu, 2019). We collected five images from each of the subjects, summing to 20 images for each of the 26 semaphore signs.

3.2. Result

For the experiments, we used half of the dataset for training and the other half for testing. We employed this method for 10 different random segmentations of the dataset for each of the features. Table II presents the performance comparison of the techniques proposed in this paper and the baseline techniques. We employed dimensionality reduction and whitening via PCA and investigated various dimension sizes. We also measured the CPU time needed for each feature extraction methodology. As expected, HOG provides a better discrimination when compared to the baseline of pure PCA on raw features.

As for the proposed, morphological processing-based methods, k-means clustering provides fast feature reduction technique that achieves good classification performance. As stated in the methodology section it suffers from the initialization ambiguities.

Table 2. Comparison of the methods proposed.

Feature	Dim	Acc (%)	CPU Time (msec)
PCA	4	81.19	9.25
	16	93.90	9.83
	32	95.03	11.52
	64	94.42	15.39
	128	93.49	20.23
HOG	4	45.07	19.85
	16	90.53	19.89
	32	95.65	19.96
	64	97.15	19.98
	128	97.30	20.19
Morph+K-means	4	96.84	9.9
Morph+Shrink	4	99.60	31.94
Morph+HAC	**4**	**99.76**	**11.75**

On the other hand, both the shrinking and the HAC-based clustering methods outperform the HOG-based features with considerably lower dimensions. With the extraction time of 31.94 milliseconds, shrinking-based clustering is impractical for online execution since the video frames are expected to be processed with 50 frames per second in real time applications. The HAC-based method, on the other hand, achieves the best performance with the fastest processing.

Figure 5 shows the average accuracy of several features with respect to their dimensionality. We see that the HOG-based methods outperform the PCA on raw images when higher dimensional vectors are used. The morphological processing-based methods, on the other hand, achieve a superior performance on the smallest dimensionality as aimed. Figure 6 demonstrates the best classification accuracy of the four feature extraction techniques with respect to the CPU time required. We see that the HAC-based method achieves the best performance with the smallest execution time.

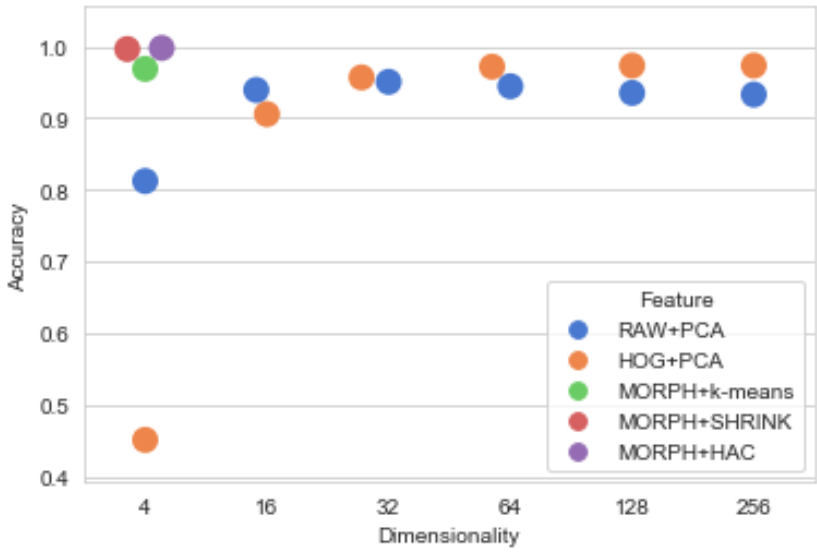

Figure 5. Classification accuracy of different methods with respect to feature dimension sizes.

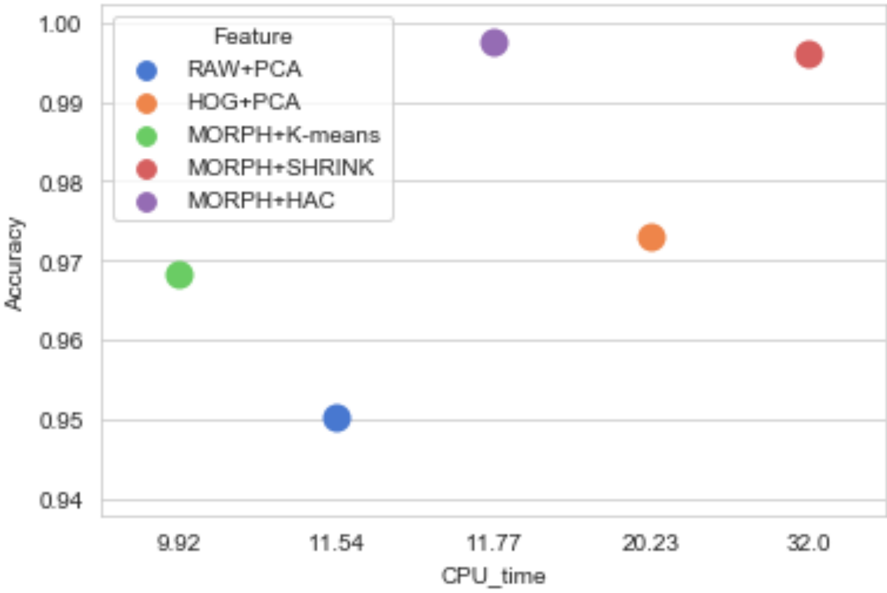

Figure 6. Classification accuracy of different methods with respect to feature extraction times.

4. CONCLUSION

In this paper, we proposed a morphological processing and clustering-based feature extraction methodology for the real-time semaphore flag signaling recognition. The red areas in the semaphore flags were detected and the centroids of the flags are calculated with three clustering methodologies. We compared the proposed methods with the HOG and PCA-based baselines and we showed that the proposed methods could be used as a high-performance low-cost feature extraction technique for the SFS classification tasks.

ACKNOWLEDGEMENT

The authors would like to thank all midshipmen of Naval Academy who took part in generating the semaphore dataset.

REFERENCES

Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., and Sheikh, Y. (2018). "OpenPose: Realtime Multi-person 2D Pose Estimation Using Part Affinity Field". *arXiv Preprint.* arXiv:1812.08008. Retrieved from https://arxiv.org/pdf/1812.08008.pdf

Dalal, N., and Triggs, B. (2005). "Histograms of Oriented Gradients for Human Detection". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05),* Vol. 1, pp. 886-893.

Day, W. H., and Edelsbrunner, H. (1984). "Efficient Algorithms for Agglomerative Hierarchical Clustering Methods". *Journal of Classification*, Vol. 1(1), pp. 7-24.

Freeman, W. T., and Roth, M. (1995). "Orientation Histograms for Hand Gesture Recognition". *International Workshop on Automatic Face and Gesture Recognition*, Vol. 12, pp. 296-301.

Gundogdu B. (2019). "Semaphore Flag Signalling Dataset". *Mendeley Data*, V1, doi:10.17632/tc5tnchrs2.1.

Hsieh, S. W., and Shih, Y. (2015). "Using Bioloid Robots as Tangible Learning Companions for Enhancing Learning of a Semaphore Flag-signaling System". *The Asian Conference on Education International Development Official Conference Proceedings.*

Hung, I. C., Hsu, H. H., and Chen, N. S. (2015). "Communicating through Body: A Situated Embodiment-based Strategy with Flag Semaphore for Procedural Knowledge Construction". *Educational Technology Research and Development*, Vol. 63(5), pp. 749-769.

Ikizler, N., and Duygulu, P. (2009). "Histogram of Oriented Rectangles: A New Pose Descriptor for Human Action Recognition". *Image and Vision Vomputing*, Vol. 27(10), pp. 1515-1526.

Iwane, N. (2012). “Arm Movement Recognition for Fag Signaling with Kinect Sensor”. *IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS)*, pp. 86-90.

Kara, Y. A., Uçarer, Ö. K., and Gündoğdu, B. (2019). “Automatic Warship Recognition System: Dataset, Feature Representation and Classification Analysis”. *27th Signal Processing and Communications Applications Conference (SIU)*, pp. 1-4.

Lloyd, S. (1982). “Least Squares Quantization in PCM”. *IEEE Transactions on Information Theory*, Vol. 28(2), pp. 129-137.

Motty, A., Yogitha, A., and Nandakumar, R. (2019). “Flag Semaphore Detection Using Tensorflow and Opencv*”. International Journal of Recent Technology and Engineering*, Vol. 7(6).

Müllner, D. (2011). “Modern Hierarchical, Agglomerative Clustering Algorithms”. *arXiv preprint.* arXiv:1109.2378.

Pratt, William K. (2013). *Introduction to Digital Image Processing*. CRC Press.

Rachmad, A., and Fuad, M. (2015). “Geometry Algorithm on Skeleton Image based Semaphore Gesture Recognition”. *Journal of Theoretical and Applied Information Technology*, Vol. 81(1), pp. 102.

Rao, C. K., Prasada, B., and Sarma, K. R. (1976). “A Parallel Shrinking Algorithm for Binary Patterns”. *Computer Graphics and Image Processing*, Vol. 5(2), pp. 265-270.

Suykens, J. A., and Vandewalle, J. (1999). “Least Squares Support Vector Machine Classifiers”. Neural Processing Letters, Vol 9(3), pp. 293-300.

Temlioglu, E., Erer, I., and Kumlu, D. (2017). "Histograms of Dominant Orientations for Anti-personnel Landmine Detection Using Ground Penetrating Radar". *4th International Conference on Electrical and Electronic Engineering (ICEEE)*, pp. 329-332.

Tian, N., Kuo, B., Ren, X., Yu, M., Zhang, R., Huang, B., and Sojoudi, S. (2018). "A Cloud-based Robust Semaphore Mirroring System for Social Robots". *IEEE 14th International Conference on Automation Science and Engineering (CASE)*, pp. 1351-1358.

Wikimedia Commons (n.d.). "Semaphore Signals A-Z". Retrieved from https://commons.wikimedia.org/wiki/File:Semaphore_Signals_A-Z.jpg

Zhao, Q., Li, Y., Yang, N., Yang, Y., and Zhu, M. (2016). "A Convolutional Neural Network Approach for Semaphore Flag Signaling Recognition". *IEEE International Conference on Signal and Image Processing (ICSIP)*, pp. 466-470.

Zhu, Q., Yeh, M. C., Cheng, K. T., and Avidan, S. (2006). "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2, pp. 1491-1498.