A New Traffic Sign Recognition Technique Taking Shuffled Frog-Leaping Algorithm into Account

Pouya Demokri Dizji[1], Saba Joudaki[2*], Hoshang Kolivand[3]

[1][2] Department of Computer Engineering, Khorramabad Branch, Islamic Azad University, Khorramabad, Iran

[3]Department of Computer Science, Liverpool John Moores University, Liverpool L3 3AF, UK

[*]Corresponding Author: Saba Joudaki, joudaki@khoiau.ac.ir

Every day human use cars to move faster, and the world is a chaotic place, and a little distraction or mistake could be the reason for an accident and bring people a great pain. With the help of computers, this can be easier. An assistance system that can distinguish and detect signs on the roads and bring the driver's attention to road signs and make them aware of their meaning could be beneficial. The most important part of the Traffic Sign Recognition System is the algorithm. In this paper a new way toward Traffic Sign Recognition algorithm taking the advantages of Color Segmentation, Support Vector Machines (SVM) and Histograms of Oriented Gradients (HOG) on the GTSRB dataset is proposed. The unsupervised Shuffled Frog-Leaping (SFL) algorithm is employed for segmenting the images. Results are showing improvements by using meta-heuristic algorithms.

Keywords: TSR, SFLA, HOG, Unsupervised Segmentation, SVM

I. Introduction

Traffic signs are vital to drivers. They control the stream of traffic, present the right of way, notify a driver about the routes and distances, supervise a destination, or warn them in unsafe places. By raising the number of drivers, there is a need to make roads safer. Advanced driver assistant systems could be the solution for this kind of problem. One of this system's parts is a traffic sign recognition algorithm that can recognize essential traffic signs for drivers. At the time of drafting this paper, a few car manufacturers have deployed commercial TSR systems in their vehicles. The TSR system is capable of notifying drivers in certain circumstances, e.g., if the car exceeded the speed boundary or if it was getting close to a junction. The variety of colors and shapes in traffic signs in various locations could cause a mismatch in TSR system results.

The sign detection is fundamental for TSR systems. Sign detection can detect signs in photos based on the color, size, shape, or other details of signs. However, because of the change of lights and view angles or the object being in front of the sing, it cannot always rely on such features, and the proposed detection process will fail as Figure 1 shows. In this paper, the authors encourage an augmented algorithm for TSR, which connects profits of the former algorithms plus using of Shuffled Frog-Leaping Algorithm. The offered method holds a more reliable production with previously given algorithms that used Conventional Image Segmentations or detection based only on the shape, color, or other simple features.

Figure 1 some easy to find and hard to find signs

The primary purpose of traffic signs is to be quickly identified and understood in the eyes of drivers. They obey a sharp teaching scheme using colors, appearance, symbols, and indications. Those provide for a wide range of distinctions among other signs. Signs that have the same meaning, like the speed boundaries, have a typical generic face, so they are very similar to each other, just drawn different numbers on them. Humans have approximately excellent accuracy for recognizing the enormous diversity of real road signs in many situations. However, the sign classifier has to overcome various visual expressions like lighting diversity, partial visibility, angles, and weather circumstances. Ordinarily, clipped images are the input for TSR system training and testing purposes, but in the natural environment, perspective might be limited based on cameras' count or dimensions, so the TSR system should present a magnificent job analyzing the inputs. Of course, providing better inputs could be very helpful.

II.   Related Work

Remarkable researchers offered concepts for better traffic sign recognition. Ellahyani et al., [1] selected HSV for Color Space and applied enhancement via a predefined threshold. Then, they did shape classification for TSR. For the test dataset that used photos of the STSD, it also generated excellent outcomes.

Novak et al., [2] created an engine using YOLO augmented including a CNN for TSR. Because real-time is vital for driving, their paper introduces the TSR system which is tweaked to recognize just five intentions grouped in classes like traffic signs, cars, pedestrians, traffic lights, and trucks. Their CNN has the capability of classifying 75 traffic sign classes. The results produced accurately were more significant than 99.2%.

Stallkamp et al., [3] offer a compact Traffic Sign dataset containing higher than 50,000 images taken from German road signs within 43 sign categories and is available to the public. Their data estimated the 2nd of the GTSRB in IJCNN 2011.

Maldonado-Bascón et al., [4] created a TSR taking SVM into account. The algorithm contains three stages, first segmentation by the color of pixels, then sign detection based on appearance via linear SVMs classification. Finally, content identification using Gaussian kernel SVMs. For detecting traffic signs with any color, they used the segmentation stage concerning single colors and combinations of colors. Experimental results showed that the system has excellent accuracy and resistance to angle rotations plus dimension changes, various positions, and even partially open signs.

Sermanet, LeCun [5] employed CNN for the TSR system. They modified the architecture of traditional CNN by feeding first and second stage features to the CNN classifier. Their crafted system produced an accuracy of 98.97%.

Sermanet et al., [6] offer a framework for handling CNNs for classification, localization, and detection, and they also deliver a feature extractor model named Overeat. They witnessed that when they use CNNs with a sliding window, it has better outcomes. They showed a CNN that can identify an object's bounding box unitedly among its class names.

Stallkamp et al., [7] manifested the study and review of the GTSRB dataset. They presented the competition of GTSRB. That is a multi-category classification contest taken at IJCNN 2011. Furthermore, the participants produced an excellent accuracy of higher than 98.98% accurate recognition percentage comparable to human achievement on this dataset.

Jain et al., [8] suggest a unique deep learning method using the idea of domain shift training for the TSR. Their method employs a recently introduced modification of the GA for determining the optimal amounts of the epoch count, each learning rate for layers of the pre-trained CNN model, and achieved correctness of 99.16% for BTSC and 96.28% for TT100K datasets.

This research presents a TSR approach by implementing the Shuffled Frog-Leaping Algorithm, being the image segmentation part of the proposed algorithm. This method contains three steps, which it can briefly explain as normalization, Detection including SLFA and SVM, and last but not least, recognition by CNN. First, by normalizing the input image, color change effects are crafted by whether and the result will narrow down by color and SFLA segmentation. Size and shape will pick candidates from remains, and a Pre-Trained SVM with Sign and Non-Sign images will remove unrelated objects for better accuracy. A convolutional neural network trained by the GTRSB dataset will classify the specific type of sign and, in the end, will return the position and name of the recognized traffic signs in output.

III.   Datasets

The proposed method's primary dataset for CNN training and testing purposes is the GTSBD and the GTSRB datasets, which are publicly available for use. These datasets are commonly used for TSR training and testing, and the reason is the similarity with the natural world based on light and angles and details like the presence of other things like trees. The GTSBD "German Traffic Sign Benchmark" Dataset Contains above 50,000 images and 43 different types. Some examples are viewable in Figure

2. This paper used 40,000 database examples for training and the remains left for testing purposes [3]. The GTSRB [3] "German Traffic Sign Detection Benchmark" Dataset contains more than 50,000 images and 43 different species. An example of the GTSRB database is viewable in Figure 3. Another essential part of the proposed method is SVM. Its job is to decide the detected ROIs by the last step whether they contain a Traffic Sign or not, and based on that decision, that ROI would get transferred to the recognition part. This task helps the algorithm significantly to decrease the amount of incorrect data fed to the recognition step. The detection step and categories generate the dataset crafted for this section. Therefore, signs are one, and no-sign is zero.

Figure 2 Some of the Images in GTSBD Dataset

Figure 3 One of GTSRB database images

IV.   Image Normalization

The goal of Normalization is to advance the picture to a scale that is standard to sense. Image normalization is the method that improves the area of pixel depth amounts. The photos might have come with some problems like low contrast due to glare. The RGB Normalization was used to get better results [21], and then the colors are much easier to filter. For Normalization a photo, the Normalized formula for an RGB pixel equals red part is (1) and the formula for the blue part is (2), and finally for the Green part used (4). An example run in a photo is viewable in Figure 4, (a) (a) is the first image without any changes, and Figure 4 (b) results after the normalization process on that image. There is a noticeable range of colors in the resulting image, which will be more simplified than the original.

$$R_p = \frac{R_p}{R_p + G_p + B_p} \tag{1}$$

$$B_p = \frac{B_p}{R_p + G_p + B_p} \tag{2}$$

$$G_p = \frac{G_p}{R_p + G_p + B_p} \tag{3}$$

(a)   (b)

Figure 4 Road Photo Normalization

V. Localization and Detection

Image Segmentation plays an essential role in image processing methods making detection more effortless by splitting images to object pieces based on features or color changes. Some popular image segmentation methods involve some edge detection solution based on clustering rules, region-based and weak supervised learning especially in Convolution Neural Network (CNN) [10]. This paper presents SFLA as an image segmentation method toward better results.

Image Segmentation by SFLA

Shuffled frog-leaping algorithm which is known as SFLA is a search method based on population. SFLA has the capability of exchange global information and local search for data. This method contains a frog community which interacts with each other and grouped in meme complexes. They act as owners or bearers of memes, wherever a meme is a unit of developmental growth. The proposed algorithm performs the individualistic local search in any memeplex concurrently. [11].

SFLA is a meta heuristic scheme that simulates a collection of frogs' memetic development as investigating the area with the highest volume of possible meals. The fogs are accepted as hosts for memes and report as a memetic vector with equal construction. However, unlike flexibilities, they can talk with anyone and enhance their memes by turning on each other. In the SFLA algorithm for optimization difficulties, any frog joined with its flexibility, determined by a unique fitness, commonly describes a possible solution. The partitioned solution divides frogs into many subgroups and mentions them as memeplexes. The diverse memeplexes took as different cultures of frogs. Then, frogs in any memeplex present a local hunt based on particular plans to provide meme transfer capability. Following a predefined amount of memetic development levels, passing data between memeplexes in a shuffling will occur. The local search also shuffling manners are conducted alternatively until the defined convergence model is fulfilled [12].

The SFLO algorithm describes the following expression. Expressly, consider that the initial community crafted by F generated random frogs. $X\_i, i=1,2,3,4,…,F$. The fitness function of X_i described by $⟦f(X)⟧\_i$ is used to assess the frog's efficiency. Moreover, one $\overline{X}$ order is created for sorting all frogs as descending arrange and also it is divided into m memeplexes, $Y^1, Y^2…,Y^m$ based on (4), subsequently, each memeplex have n frogs [12], which is a m×n function.

$Y^K=\{X\_i^k \mid X\_i^k = X\_{(K+m(i-1))}\}, i=1,2,3,4,…,n.\text{and } k=1,2,3,4,…,m.\}$ (4)

Inside any memeplex, frogs are the most reliable and most unfavorable fitness distinguished as X_b and X_w, respectively. In addition, the X_g parameter is a frog by the best fitness among all existing frogs. The nearby advancement look is given out in correspondence in each memeplex to alter concurring to (5) (6) [12].

$$D = rand() \times (X_b - X_w) \qquad\qquad (4)$$

$$X'_w = X_w + D, D_{min} \leq D \leq D_{max} \qquad\qquad (5)$$

In equation (5) rand () function is a random generated number between 0 and 1. Furthermore, the limit of change in the frog's position are shown as D_min and D_max numbers for the minimum and maximum of its location. If this procedure leads to a better frog (X_w^'), the worst one (X_w) is replaced. On the other side, if it does not lead to a better position, the global best frog (X_g) is used to replace (Xb) to perform the mentioned rules. A feasible solution is randomly generated to replace if there is no improvement yet. Calculations are ongoing for a predetermined number of iterations in each memeplex, memeplex. In addition, the entire population is involved in the process of displacement (shuffling). Local exploration and world displacement alternate until a predefined convergence requirement is met. Kapur et al (1985). The entropy factor was broadly utilized in deciding the ideal threshold for segmenting the images. The first calculation had been created for bi-level edge. The strategy can moreover expand to illuminate multilevel edge issues and can be portrayed as takes after. Let there be "L" gray levels in a given picture and these gray levels are in the desired range [12].

It can be reffered P_i=h (i)/N,(0≤i≤L-1) where:

The $h(i)$ parameter shows the number of the pixels with gray-level i and the N parameter shows the total number of the pixels in the considered image.

There is a set of D threshold $[t_1, t_2, \ldots, t_D]$ and one objective function f for a given image i as shown in (7). So, it can get the equivalent (8):

$$f([t_1, t_2, \ldots, t_D]) = H_0 + H_1 + H_2 + \cdots + H_D \tag{6}$$

$$\omega_j = \sum_{i=t_1}^{t_2-1} P_i, \quad H_j = -\sum_{i=t_j}^{t_{(J+1)}-1} \frac{P_i}{\omega_j} \ln \frac{P_i}{\omega_j} \tag{7}$$

The proposed MESFLOT method seeks to achieve this excellent D-dimensional vector $[t_1,t_2,\ldots,t_D]$, which is capable of maximizing. Additionally, for the specific fitness of the MESFLOT method, the objective function is needed. The used method is a segmentation solution based on SFLA optimization by the greatest entropy thresholding method. [12].

The image segmentation is vital for the image processing system. It is needed before analysis of the images at the junction between the analysis and processing. An unsupervised segmentation purposes to separate the images in some clusters automatically [13]. By the acquired photo from the Pre-Processing step in Figure 5 (a), the image was segmented with SFLA and only kept the important part by the colors as shown in Figure 5(b). Then some noises were removed with Gaussian filter Figure 5(c).
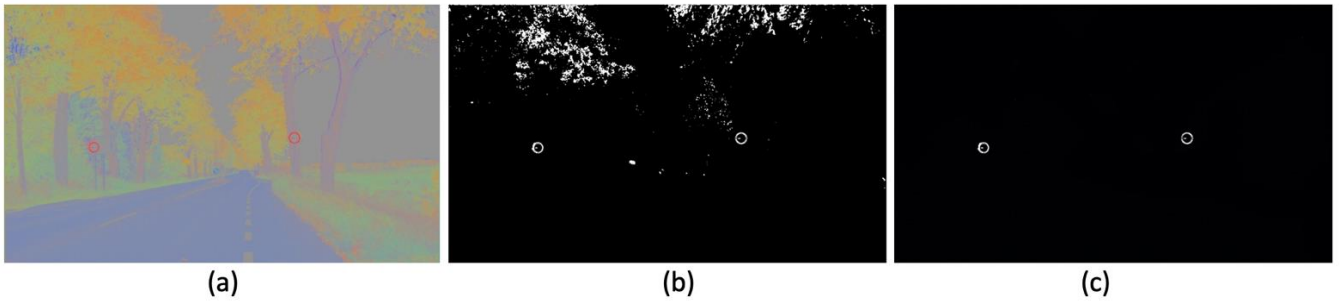

(a)   (b)   (c)

Figure 5 Image Segmentation Process.

Region of interest

An essential part of image analysis is image segmentation. For object detection in the image, ROI discovery is applied in many areas. Classification of the ROI of an image can be performed using a manual or automatic process. There have been many kinds of research to extend automatic ROI discovery by using picture segmentation methods. The depth of image segmentation is based on the final answer that is to be applied in the employment. There are several complex methods that had been suggested for identifying ROI. Recognizing the object or region of interest (ROI) in a real view is difficult since the content of raw images comprises of several non-uniform sub-regions and the severity in-homogeneities [14]. Figure 6 (b) shows the ROI detection from the OpenCV library.
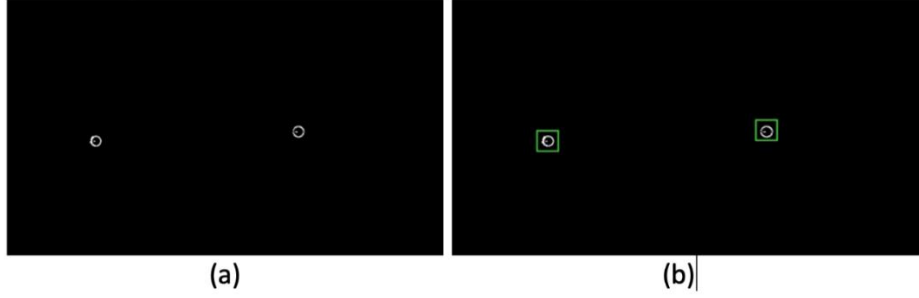
Figure 6 ROIs detected from Image.

VI. Linear SVM

For checking the previous step's results to detect the sign, the authors are using a Linear SVM. This classifier only expresses which whether the input image is a sign or not, and then the classify step is begun by neural network. ROIs detected in the last part might be contained with some unrelated objects, and in this step they will be removed from unrelated objects as shown in Figure 7. The SVM will approach classifying using appearance features, that process accepting Histogram of Oriented Gradients (HOG) as input data. HOG is a highly effective feature descriptor for making the job completed [15]. For training Linear SVM, the HOG (Histograms of Oriented Gradients) feature is used to create a vector of images that can be used for training [16] an example is viewable in Figure 8. For Linear SVM training, the authors used 1000 image samples for output 1, positive sign detection and 1000 background images for output 0 for negative sign detection.
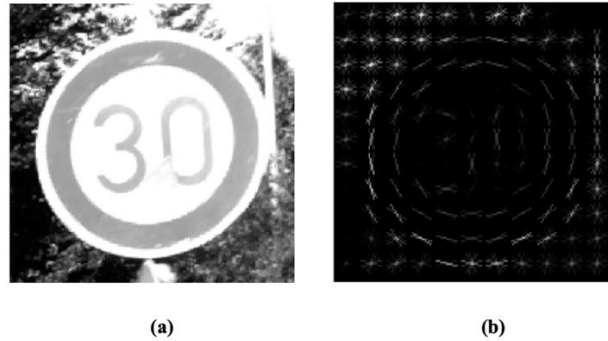

Figure 7 Images from ROIs detected from Image.

Figure 8 Sign image for extracting HOG futures.

VII. Classification and Recognition

An advance to the construction of high-performance image classification systems is possible by combining a sum of neural networks. Neural network and its family have obviously a robust learning methods [9]. It often has been proven that the previous researchers benefit from using a Natural Network in the pattern recognition to improve image classification systems. Nevertheless, earlier work noted that this kind of image classification method is decent only if the neural networks producing them make various errors [17]. The ANN is widely used for sign recognition but the light conditions significantly effect on the detection process [18]. DNNs plus Deep Learning are strong and approved methods in this kind of case. Moreover, a lot of their

victory rests in carefully creating the CNN structure. In Table 1, the architecture details for CNN are viewable. The input has three layers of 64 x 64 arrays because the considered images will resize to a 64 x 64 arrays with three-dimensional dimensions and each domain belongs to a color. The convolutional neural network uses a sequence of layers. The type of layer is what this layer will do. The convolution layer is needed for extracting spatial features from an image. Max pooling is a sample-based discretization process that takes the maximum of that region and creates an output matrix

Figure 9 Sign Classes in the dataset

The convolutional neural networks are a type of multilayer perception. It derives its name from the convolution operator. CNN is frequently used in image processing, audio processing, natural language processing, and biomedical. CNN provides the most successful results in the fields of image processing, image recognition, and image classification. Convolution's primary purpose is to extract the features of the input image. It uses small squares of convolution input data. In this way, it maintains the spatial connection among pixels by discovering the specialties of pictures. The layers that make up the architecture of convolution neural networks are given as subheadings [1].

Table 1 Architecture of Neural Network used for classification

Layer id	Layer Type	#	Map	Kernel
0	**Input**	**3**	**64 x 64**	
1	**Convolutional**	**64**	**60 x 60**	**3 x 3**
2	**Max pooling**	**64**	**60 x 60**	**2 x 2**
3	**Convolutional**	**128**	**30 x 30**	**3 x 3**
4	**Max pooling**	**128**	**30 x 30**	**2 x 2**
5	**Dense**		**512**	**1 x 1**
6	**Dense**		**43**	**1 x 1**

The result of neural network training is shown in Figure 10. Accuracy is a metric that describes the percentage of the test data that is classified correctly. A great accuracy with low loss means you made low errors on data. The lower the loss, the better a model. In **Error! Reference source not found.**, the test process on the number of model's accuracy and losses on the Train and Test dataset can be seen.
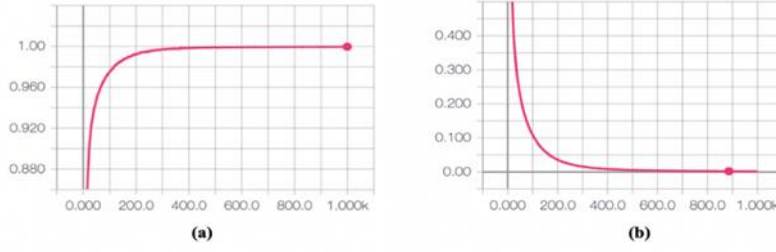
Figure 10 Accuracy and Losses

VIII.   Experimental Results

After running the algorithm in a normal laptop, results have been promising, without any special hardware, this system runs with a good FPS and good accuracy. This leads to high hopes for the ability to use this algorithm on mobile devices. In Figure 11 some photos are shown with different kinds of lighting and angles and also the different counts of signs in the photo. The authors compared the result of the proposed algorithm to another algorithms reported by the owner of the GTRSBD database in Table 2 [3].

Figure 10 Some Results of Algorithm

Table 2 A compare between the previous methods and the proposed method

	Method Name	Team	CCR (%)
	Human (Best individual) [2]	INI-RTCV	99.22
	Human (Average) [2]	INI_RTCV	98.84
	Multi-Scale CNN [3]	Sermanet	98.31
	LDA (HOG 2) [2]	INI_RTCV	95.68
	LDA (HOG 1) [2]	INI_RTCV	93.18
	LDA (HOG 3) [2]	INI_RTCV	92.34
*	**SLFA_HOG_CNN**	**Demokri et al.**	**99.69**

IX. Conclusion

The most important part of a TSR Algorithm is the Detection part. Without reliable results from this step, the rest of the steps can't perform their assignments accurately. We can say the detection part is the crucial ingredient. Considering SLFA for making this part perfect, it did an excellent job decreasing errors and improving accuracy. The improved algorithm presents profitable outcomes.

This paper introduces a complete and complex method for traffic sign recognition. The proposed method contains three steps: image pre-processing, detecting the location of traffic signs using SFLA for image segmentation and taking benefit for tracking signs and a SVM to approve outcomes, and lastly, recognizing traffic sign classes using CNN. The SFLA was a reliable way for better-unsupervised image segmentations. This adjusted algorithm helps to remove noise, avoid false discovery of signs, and stimulate image processing. The revealed algorithm can increase the quality, enhance the reliability of TSR methods, and overcome the time needed for real-time processing the frames.

The presented algorithm showed promising results with high accuracy. However, these numbers drop when encountering different lighting and angles. Datasets with more light changes could achieve better TSR recognition models. Other meta-heuristic methods like Glowworm swarm, Grey wolf optimizer, and Multi-verse optimizer study for better future work results.

References

[1] I. CINAR, Y. S. TASPINAR, M. M. SARITAS and M. KOKLU, "FEATURE EXTRACTION AND RECOGNITION ON TRAFFIC SIGN IMAGES," *Journal of Selcuk-Technic Özel Sayı 2020 (ICAT'20) Special,* pp. 282-292, 2020.

[2] J. Stallkamp, M. Schlipsing, J. Salmen and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks,* 2012.

[3] P. Sermanet and Y. LeCun, "Traffic Sign Recognition with Multi-Scale Convolutional Networks," in *The 2011 International Joint Conference on Neural Networks*, 2011.

[4] S. Yuheng and Y. Hao, "Image Segmentation Algorithms Overview," *CoRR,* vol. abs/1707.02051, 2017.

[5] A. Tungkasthan and W. Premchaiswadi, "Automatic Region of Interest detection in natural images," 2011.

[6] A. Tehami and F. Hadria, "Unsupervised Segmentation of Images Based on Shuffled Frog-Leaping Algorithm," *JIPS,* vol. 13, pp. 370-384, 2017.

[7] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou and K. Q. Weinberger, Eds., Curran Associates, Inc., 2012, pp. 1097-1105.

[8] G. D. Finlayson, B. Schiele and J. L. Crowley, "Comprehensive colour image normalization," in *Computer Vision --- ECCV'98*, Berlin, 1998.

[9] H. Kolivand, S. Joudaki, MS. Sunar & D. Tully, "A new framework for sign language alphabet hand posture recognition using geometrical features through artificial neural network (part 1)," *Neural Computing and Applications,* https://doi.org/10.1007/s00521-020-05279-7, 2020.

[10] A. Ellahyani, M. E. Ansari and I. E. Jaafari, "Traffic Sign Detection and Recognition Based on Random Forests," *Appl. Soft Comput.,* vol. 46, pp. 805-815, 9 2016.

[11] R. Ebrahimzadeh and M. Jampour, "Efficient Handwritten Digit Recognition based on Histogram of Oriented Gradients and SVM," *International Journal of Computer Applications,* vol. 104, pp. 10-13, 10 2014.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[13] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li and S. Hu, "Traffic-Sign Detection and Classification in the Wild," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, USA, 2016.

[14] S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno and F. Lopez-Ferreras, "Road-Sign Detection and Recognition Based on Support Vector Machines," *IEEE Transactions on Intelligent Transportation Systems,* vol. 2, pp. 264-278, 8 2007.

[15] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," *CoRR,* 2013.

[16] J. Stallkamp, M. Schlipsing, J. Salmen and C. Igel, "The German Traffic Sign Recognition Benchmark: A multi-class classification competition," in *The 2011 International Joint Conference on Neural Networks*, 2011.

[17] A. Jain, A. Mishra, A. Shukla and R. Tiwari, "A Novel Genetically Optimized Convolutional Neural Network for Traffic Sign Recognition: A New Benchmark on Belgium and Chinese Traffic Sign Datasets," *Neural Processing Letters,* vol. 50, no. 3, p. 3019–3043, December 2019.

[18] H. Kolivand, S. Joudaki, MS. Sunar & D. Tully, "An Implementation of Sign Language Alphabet Hand Posture Recognition using Geometrical Features through Artificial Neural Network (Part 2)," *Neural Computing and Applications,* 2021.

[19] Š. Toth, "Difficulties of Traffic Sign Recognition," *MATHEMATICS APPLIED TO ICT,* 2012.

[20] B. Novak, V. Ilic and B. Pavkovic, "YOLOv3 Algorithm with additional convolutional neural network trained for traffic sign recognition," 06 2020.

[21] M. Eusuff, K. Lansey and F. Pasha, "Shuffled frog leaping algorithm:A memtic meta heuristic for discrete optimization," Engineering Optimization - ENG OPTIMIZ, vol. 38, pp. 129-154, 3 2006.

[22] M. Horng, "Multilevel image threshold selection based on the shuffled frog-leaping algorithm," *Journal of Chemical and Pharmaceutical Research,* vol. 5, no. 9, pp. 599-605, 2013.