Neural Computing and Applications (2021) 33:11661–11672
https://doi.org/10.1007/s00521-021-05881-3

ORIGINAL ARTICLE

An advance ensemble classification for object recognition

Ebenezer Owusu[1] · Isaac Wiafe[1]

Received: 24 December 2019 / Accepted: 22 February 2021 / Published online: 11 March 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract
The quest to improve performance accuracy and prediction speed in machine learning algorithms cannot be overemphasized, as the need for machines to outperform humans continue to grow. Accordingly, several studies have proposed methods to improve prediction performance and speed particularly for spatio-temporal analysis. This study proposes a novel classifier that leverages ensemble techniques to improve prediction performance and speed. The proposed classifier, Ada-AdaSVM uses an AdaBoost feature selection algorithm to select small features of input datasets for a joint support vector machine (SVM)–AdaBoost classifier. The proposition is evaluated against a selection of existing classifiers (SVM, AdaSVM and AdaBoost) using the Jaffe, Yale, Taiwanese facial expression database (TFEID) and CK + 48 datasets with Haar features as the preferred method for feature extraction. The findings indicated that Ada-AdaSVM outperforms SVM, AdaSVM and AdaBoost classifiers in terms of speed and accuracy.

Keywords SVM · Adaboost · AdaSVM · Ada-AdaSVM

1 Introduction

Existing datasets that are used for model training are characterized by class imbalances (with some target classes containing fewer examples), noise, under-sampling and over-sampling. To address these challenges, researchers continue to propose newer algorithms that outperform existing ones. For instance, recent studies have demonstrated that distributed bandit-based algorithms reduce network data and are also capable of finding clusters faster with higher accuracies in recommender systems [1]. Others have demonstrated that performance accuracy can be improved in imbalance dataset by applying the Synthetic Minority Over-sampling Technique (SMOTE) with boosting procedures [2]. Particularly, by transforming the selection problem in SMOTE to a multi-objective optimization problem [3]. Similarly, others [4–6] have explained procedures that can be adopted to improve prediction accuracy. It has been demonstrated that ensemble provides a promising solution for training more accurate classification models. This is because, in ensembles, multiple classifiers are used to train the model. This allows each classifier to make use of the other's strengths to produce a robust model that is more accurate [4]. Currently, the two main methods of ensemble construction are based on boosting and bagging. Although, some studies have advocated for bagging as a preferred choice [7], others have argued that in cases where datasets have less noise, boosting outperforms bagging [8]. For instance, Adaptive boosting (AdaBoost) can be generalized since its margins can be enlarged and thus performs better. AdaBoost creates a collection of classifiers by maintaining a set of weights over the training data and adaptively adjust these weights during each boosting iteration.

Admits all the success, the need to achieve higher levels of classification accuracy remains. This is because the fundamental trade-off between accuracy and classification speed serves as an impediment in classification model training. As such, domains that benefit from existing machine learning algorithms continue to propose methods that seek to improve prediction accuracy. A domain that benefits most from classification algorithms and improved modeling performance is object recognition. Object recognition is relevant and has potentials in other areas of study including facial recognitions and expressions, fingerprint recognition and retina recognition. However, the

✉ Isaac Wiafe
iwiafe@ug.edu.gh

[1] Department of Computer Science, University of Ghana, Accra, Ghana

domain inherits generic machine learning model training challenges. As such, the ability to classify or detect objects more accurately with lesser computational complexity continues to be one of the main challenges in the domain. This challenge is more visible in image recognition because image datasets are characterized by several issues including pigmentation, aspect size and resolution that compounds the classification problem. In this regard, this study seeks to explore the strengths of using ensemble techniques with support vector machine (SVM) and AdaBoost to improve classification accuracy without compromising classification speed. Our motivation for using SVM and AdaBoost is derived from studies conducted by Owusu et al. [9] that demonstrated that both model performance and speed can be improved by using AdaBoost feature extraction and SVM to train a classifier. In this study, it is hypothesized that model performance and training speed can further be improved by using AdaBoost for feature extraction and a fusion of multiclass support vector machine and AdaBoost as a hybrid classifier.

In the next section, a discussion on relevant literature on algorithms for object recognition, classification and ensembles is presented. This is followed by a description of the datasets and the pre-processing stages. The proposed technique (Ada-adaSVM) is then discussed. The model evaluation and performance measure, discussions and conclusions are then presented.

2 Related work

2.1 Algorithms for object recognition and classification

Machine learning methods and algorithms have become rampant and more successful in recent times because computing hardware cost and processing times have reduced exponentially. Therefore, training complex algorithms are now relatively easier and faster. As mentioned earlier, the relevance of object recognition and image classification cannot be overemphasized. Accordingly, several studies have proposed methods that have proven to be effective with higher accuracies for classifying or recognizing images. Methods that leverage on Deep Learning algorithms, Bayesian Network, Decision Trees, *k*-Nearest Neighbors, Support Vector Machine, among others have all been used in image classification and recognition research.

Deep learning is effective in image classification as they are capable of extracting the appropriate features and also discriminate at the same time [10]. They are broadly classified into four main categories; Restricted Boltzmann Machines (RBMs), Convolutional Neural Networks (CNNs), Autoencoders and Sparse Coding. Each method is adopted or applied based on needs or expectations. They require fewer statistical training and also can detect complex nonlinear relationships between dependent and independent variables. Yet, they produce multiple solutions associated with local minima, hence they are not considered to be robust over different samples [11] or general-purpose algorithms since they require huge data to train. They operate in a "black box" and thus have a greater computational burden and are also prone to overfitting [12]. Deep learning algorithms are also outperformed by tree ensembles for even typical machine learning problems.

Deep learning methods have improved over the years [13], although its theoretical underpinnings are not well understood. Also, due to the complexity involved in training deep learning models, they are considered inappropriate for applications that run in "real-time". Nonetheless, they are considered to be one of the successful spatial techniques available [9].

2.2 SVM and AdaBoost techniques

Adaptive boosting (AdaBoost) is a spatial learning algorithm that selects a restricted number of weak classifiers from a large weak classifier hypothesis space, to construct a strong classifier. The boosting process attempts to improve the model by focusing on where it is not learning properly. It is effective for reducing bias error. However, it has been argued that although it performs better than bagging, it can overfit [14]. Yet, some studies have proven otherwise [15]. In boosting, parameters are tuned, and the algorithms are restricted to reducing the multiclass classification problem to multiple two-class problems [16–18].

SVM on the other hand provides a good out-of-sample generalization. They can be made robust by choosing appropriate generalization grades in the presence of biases among training samples. SVMs can be considered as maximal margin hyperplane classification techniques that depend on the outcomes of statistical learning theories to certify good generalization performance [19]. Yet, AdaBoost is considered to be better on generalization, and this makes it preferable for many classification problems although it is sensitive to noisy data and outliers. Some studies have argued that SVM is superior in recognition accuracies when compared to AdaBoost, nevertheless, the latter is faster in terms of model training speed [20].

Generally, there is a trade-off between performance (in terms of accuracy) and model complexity (in terms of training speed). In addressing this, different classification techniques are integrated to enhance performance and possibly training speed. As mentioned earlier, recent studies conducted by Owusu et al. [9] demonstrated that SVM and AdaBoost can be ensembled to improve model performance and speed.

2.3 Research contribution

As has been conferred in the previous section, AdaBoost performs better in terms of model training speed, yet poor in accuracy when compared to SVM. Thus, this study seeks to use AdaBoost as (1) initial feature selection algorithm and (2) amalgamate AdaBoost and SVM (AdaSVM) to form a classifier. It is therefore expected that the resultant classifier will perform better in terms of speed and accuracy.

2.4 Ensemble techniques

Research on the application of ensembles in machine learning and artificial intelligence application continues to advance as the need to discover newer approaches with competitive levels of accuracy, speed and lesser computational complexity becomes prominent. Several studies have focused on variations of SVM and AdaBoost for model performance. Some researchers have argued that due to the complexities in SVM, AdaBoostSVM ensemble may not be viable [21]. Nonetheless, studies conducted by Owusu et al. [20] and Li et al. [21] have experimented the possibilities available when SVM is ensembled with AdaBoost. Studies have concluded that AdaBoostSVM performs better when used as component classifiers [22], and it can improve classification performance on imbalanced data.

A key objective in the use of ensemble is its ability to balance bias (underfitting) and variance (overfitting). It employs the power of diversity to enhance training performance. However, evaluating resultant models from ensembles are more challenging and demands more computation than those of single models. That notwithstanding, the effort required for computation can be argued to be worthwhile as they perform better than single models. Fast algorithms (such as decision trees) benefit more than slower algorithms in ensembles. They perform better in situations where there is significant diversity between the individual base models and effective in capturing both linear and nonlinear patterns. Accordingly, this study adopts an ensemble technique to train an object recognition model. The next section is a discussion on the proposed model.

3 Proposed algorithm: Ada-AdaBoost

A major challenge in ensembles is the art of selecting the various base models to optimize model training performance and speed. The novelty and contribution of this study are in the ability to construct an ensemble that outperforms existing algorithms. The proposed procedure resigns from simple integration of data pre-processing with unchanged bootstrap sampling technique. As compared to standard bootstrap sampling, the probability of drawing different types of examples is changed. Rather, the sampling is focused on the minority class, specifically those located in challenging sub-regions of the minority class. This probability is dependent on the neighborhood class distribution of the sample as suggested.

The first part of the proposed method is an AdaBoost-based feature selection tool that selects a series of uncomplicated classifiers to form a strong classifier. This procedure speeds up detection and avoids misclassification. At each stage, feature extraction filters such as Gabor [20] and Haar [23] are treated as weak classifiers. This ensures that the AdaBoost feature selection algorithm selects the finest of classifiers and boosts the weights on the examples, to weigh the errors further. Afterwards, a subsequent filter is selected to provide the most appropriate result on the errors of the preceding filter. At each step, the selected filter is uncorrelated with the output of the preceding filter. After a cycle of training, samples are re-weighted, and the weight of the error classified samples is accentuated. The diverse weighted weak learners are merged to produce a strong feature selection hypothesis (see Fig. 1).

The second part of the technique is the classification. This involves the fusion of the multiclass SVM and AdaBoost to form a hybrid classifier. For this optimization problem, SVM with the Radial Basis Function (RBF SVM) kernel is used as a weak classifier. This weak SVM classifier is trained to produce the optimum Gaussian value for the scale parameter δ and the regularization parameter ∂. This method was adopted from [8, 18], and the decision hyperplane λ is represented by the equation:

$$\lambda = \frac{\sum\limits_{\forall i \in \phi^{(+)}} D(i).\phi_{(\mu,v,z)}}{\left\| \sum\limits_{\forall i \in \phi^{(+)}} D(i).\phi_{(\mu,v,z)} \right\|} + \frac{\sum\limits_{\forall i \in \phi^{(-)}} D(i).\phi_{(\mu,v,z)}}{\left\| \sum\limits_{\forall i \in \phi^{(-)}} D(i).\phi_{(\mu,v,z)} \right\|} \tag{1}$$

where $i \in (1, 2, \ldots, N)$ are extracted features of Gabor or Haar filters. An image I, is represented as $\Phi_i = \{(x_n, y_n)\}_{n=\pm 1}^{N}$ and z, μ, v represents the parameters of the image. The positive sets $\phi^{(+)}$ and the negative sets $\phi^{(-)}$ are denoted by $\phi^{(+)} = \{(x_n, y_n)\}_{n=1}^{N\ J} \subset \Re^J \times (\pm 1)$ and $\phi^{(-)} = \{(x_n, y_n)\}_{n=-1}^{N} \subset \Re^J \times (\pm 1)$, respectively, where x_n is the n-th data sample containing J features, and y_n is its corresponding class label.

The vectors $\phi_{(u,v,z)}$ over a distribution D are trained to find the weights of all feature vectors, $\Phi_i = \phi^{(+)} + \phi^{(-)}$. This gives us a threshold λ, which is the decision hyperplane. The positive half of λ (which is the majority

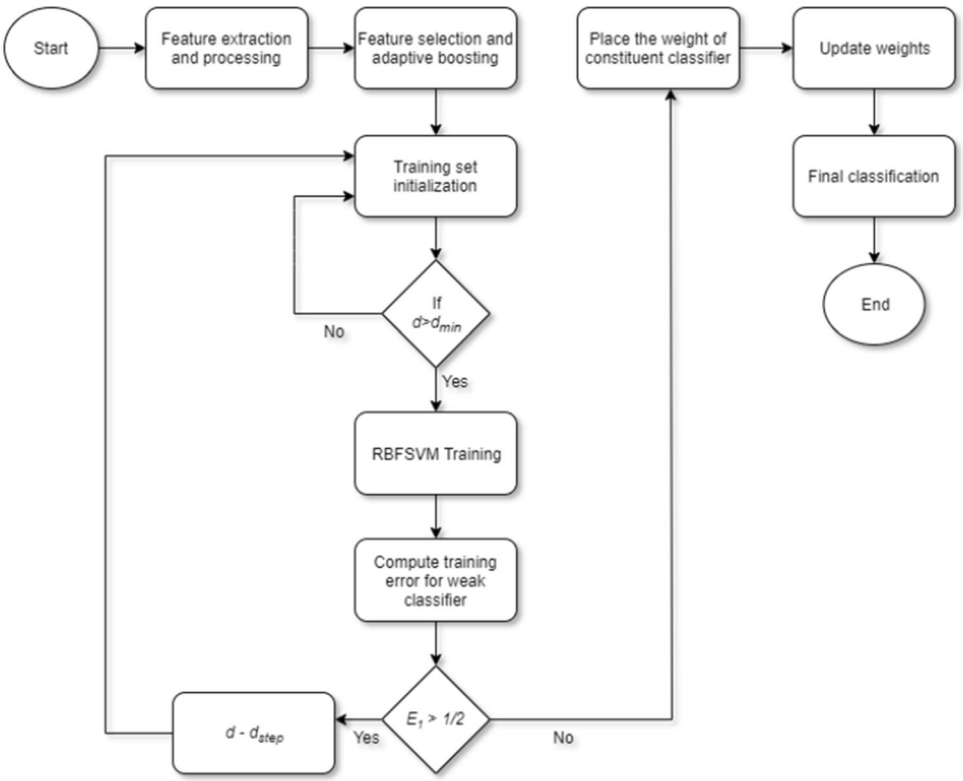

Fig. 1 Process flow of the proposed Ada-AdaSVM

decision) represents the clients, and the negative half represents the imposter. The status is reversed if the minority of the positive instances is rather located in the positive half-space. The final feature selection tool is computed as follows:

Let c be the clients and p be the imposters. For a given training dataset containing both positive and negative samples, each sample is denoted by (Φ_i, y_i), where $y \in \{\pm 1\}$ is the corresponding class label. The feature selection hypothesis $H(\Phi)$, which is a function of the selected features as suggested by [24] is formulated as below:

$$H(\Phi) = \operatorname{sgn}\left[\sum_{t=1}^{T} \omega_t h_t^1(\Phi_t^1)\right] \tag{2}$$

where T is the final iteration, h_t^1 is a hypothesis with the most discriminating information,ω_t represents the weights that weigh h_t^1 by its classification performance.

The second part of the algorithm is the learning process which is formulated as follows:

Step 1 Input the training sets, $\{(y_1, x_1), \ldots, (y_N, x_N)\}$, $N = a + b$; where datasets have $y_i = +1$ and b datasets have $y_i = -1$. The initial $\delta = \delta_{\text{ini}}, \delta_{\min}, \delta_{\text{step}}$. The scale parameter, δ and x and y are the feature vectors selected by the AdaBoost algorithm.

Step 2 Initialize the training sets weights as:

$$w_i^{(1)} = \frac{1}{2a}, \forall(y_i = +1) \quad \text{or} \quad w_i^{(1)} = \frac{1}{2b}, \forall(y_i = -1). \tag{3}$$

Do while $\delta > \delta_{\min}$.

Step 3 Apply the RBFSVM kernel to train the weighted training data sets by applying the Leave-One-Subject-Out Cross-Validation (LOSOCV) approach and compute the training error for the weak classifier h_t:

$$\xi_t = \sum\nolimits_{i=1}^{N} w_i^t, y_i \neq h_t(x_i) \tag{4}$$

Step 4 At $\xi_t > \frac{1}{2}$, reduce δ by a factor of and δ_{step} then jump to Step 1.

Step 5 Place the weight of the constituent classifier h_t such that:

$$h_t : \alpha_t = \ln\left[\frac{1}{\xi_t} - 1\right]^{\frac{1}{2}} \tag{5}$$

Step 6 Update the weights by computing:

$$w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}}{N_t}, \tag{6}$$

where N_t is a normalization constant and $\sum_{i=1}^{n} w_i^{t+1} = 1$.

Step 7 The final classifier is given by:

$$H(x) = \text{sgn}\left[\sum_{t=1}^{T} \alpha_t h_t(x)\right]. \quad (7)$$

The leave-one cross-validation approach is used to partition datasets into training and testing. The leave-one-out error E is determined by:

$$E = {}^{1}/_{2n} \sum_{i=1}^{n} |f_i(x_i) - l_i| \quad (8)$$

where n is the number of times data is trained.

4 Datasets

The proposed model was evaluated using the Jaffe, Yale, the CK + 48 and the Taiwanese facial expression database (TFEID). The Jaffe database contains 213 images of 10 female Japanese persons. Two (2) images of each individual from each class of expression namely neutral, sad, fear, anger, disgust, happy and surprise were randomly selected for training; leading to a total of 140 images (65.7%), and the remainder was used for testing.

A tenfold cross-validation was used to obtain an average recognition rate for the dataset. The Yale facial expression database contains 165 grayscale images in GIF format of 15 individuals. One hundred and thirty (130) images corresponding to happy, neutral, sad and surprise were manually extracted. The dataset was partitioned into training and testing using the same procedure for the Jaffe dataset. This resulted in 77% of the images for training and the remainder for testing. The CK + 48 comprises 981 grayscale datasets of seven facial expressions, namely, anger, contempt, disgust, fear, happy, sadness and surprise. The composition of the images is anger, 135; contempt, 54; disgust, 177; fear, 75; happy, 207; sadness, 84; surprise, 249. Over seventy-one per cent (i.e., 71.4%) of the datasets comprising 700 subjects were used for training and the remaining 29.6 comprising of 281 for testing. TFEID consists of 7200 stimuli, captured from 40 models (20 males), each with eight facial expressions: neutral, anger, contempt, disgust, fear, happiness, sadness and surprise. Models gazed at two different angles (0° and 45°). Each expression includes two kinds of intensities (high and slight) and was captured by two CCD-cameras simultaneously with different viewing angles (0° and 45°). Seventy per cent (70%) of the TFEID comprising of 5040 were used for training, while the remaining 30% (2160) were used for testing. Figure 2 shows samples of the four different datasets.

All the datasets above were processed using the following steps:

Step 1 All images were resized to a window of size 25 × 15, and RGB images (the case of TFEID) were converted into grayscale images.

Step 2 Feature extraction by Haar, H was processed using:

$$H = \frac{1}{N(u,v).\sigma_B} \sum_{i=1}^{N_B} [Sgn(B_i).S(B_i)] \quad (9)$$

where σ_B is the variance for pixel feature block B, N_B is the number of subblocks, B_i is the sign of the coefficients to the subblock (refer to [23] for more detail).

Step 3 Images were downsampled using the Bessel method [23], and it is calculated as:

$$x_d(t_1,t_2) = \sum_{m_1=1}^{a} \sum_{m_2=1}^{b} c(m_1,m_2) J_0\left(\frac{\alpha_{m_1}}{a-m} t_1\right) J_0\left(\frac{\alpha_{m_2}}{b-n} t_2\right)$$
$$0 \le t_1 \le a-m,\ 0 \le t_2 \le b-n$$

where a and b are original image sizes, $(a-b) \times (b-n)$ is the downsampling image, where m and n are any positive integers such that $m < n$ and $n < b$, J_0 is a first-order Bessel function. The size of the original images is reduced to one-fourth of the original.

Step 4 Normalization was performed using anisotropic smoothing, and the normalized image $I(x^1, y^1)$ is computed as:

$$I(x^1, y^1) = \text{scale}(\alpha R(x,y) + \text{Heq}(L(x,y)))$$

where

$$R(x,y) = I(x,y)/L(x,y),$$

Heq represents histogram equalization with $\propto = 2$.

5 Results and analysis

The computed results from the respective confusion matrices (Figs. 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18) indicate that the recognition accuracy of the proposed technique (Ada-AdaSVM) is superior to the others. The recognition accuracy-performance analysis (Figs. 19, 20, 21, 22) demonstrates that the propsed method (Ada-AdaSVM) outperformed all the other methods tested (i.e., AdaSVM, AdaBoostSVM and SVM) on all the datasets. The performance test conducted showed that the performance of Ada-AdaSVM is statistically significant ($p < 0.05$).

5.1 Computations of recognition accuracies

The recognition accuracy of all the methods is computed using a confusion matrix. The results are displayed in

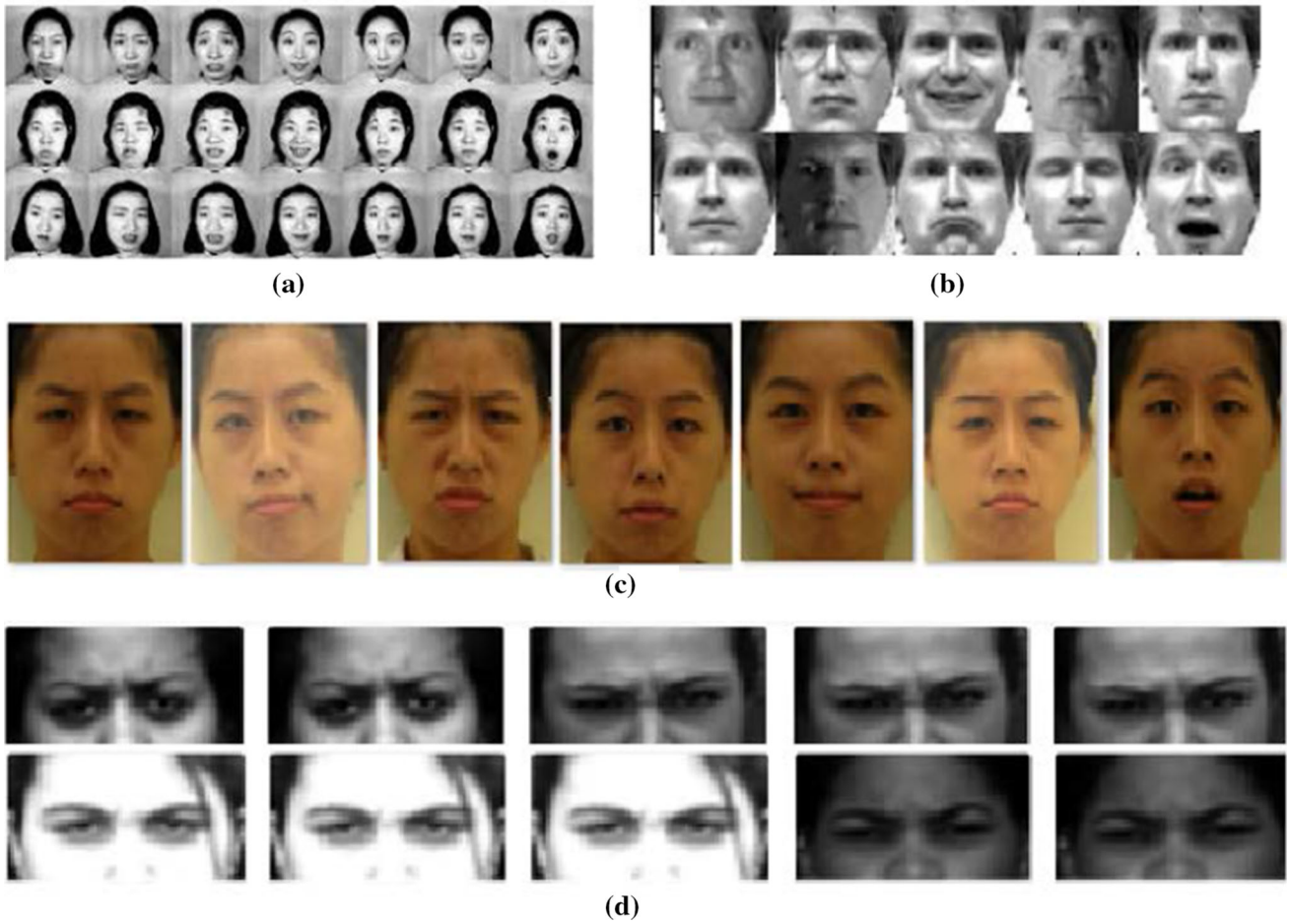

Fig. 2 Sample images in Jaffe (**a**), Yale (**b**), TFEID (**c**) and CK + 48 (**d**)

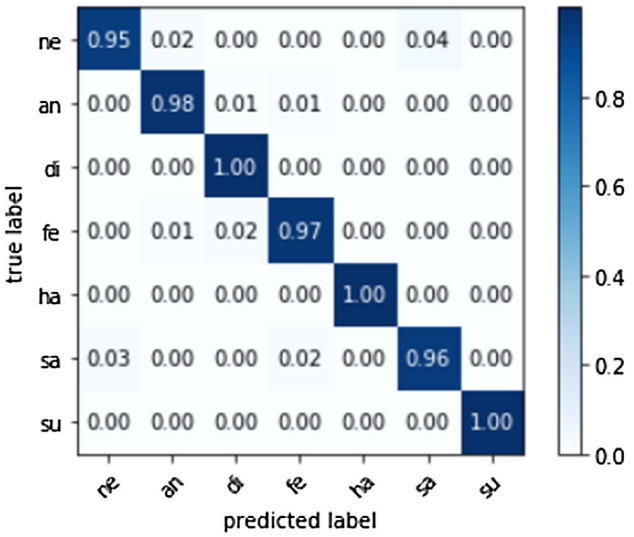

Fig. 3 Confusion matrix for Ada-AdaSVM (JAFFE dataset)

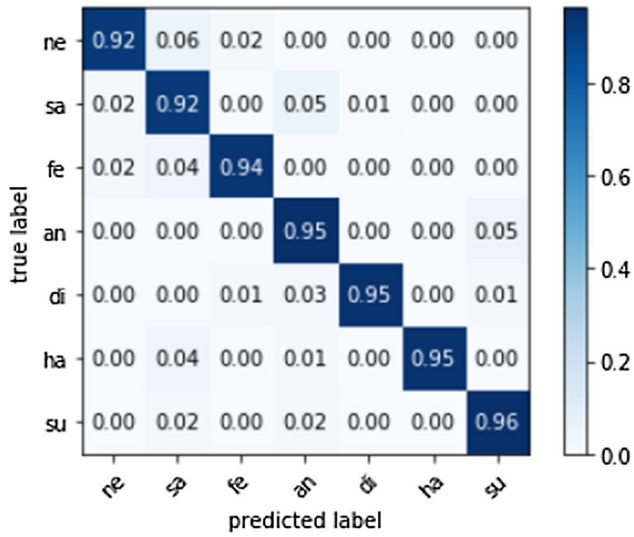

Fig. 4 Confusion matrix for SVM (JAFFE dataset)

Figs. 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 and 18 (where *neutral* = *nu*, *anger* = *an*, *disgust* = *di*, *sad* = *sa*, *surprise* = *su*, *fear* = *fe*, *happiness* = *ha*). With regard to the JAFFE dataset, it can be observed that Ada-AdaSVM outperforms all the other methods. For instance, it recorded a true positive rate of 0.95 for neutral, 0.98 for anger, 1.00

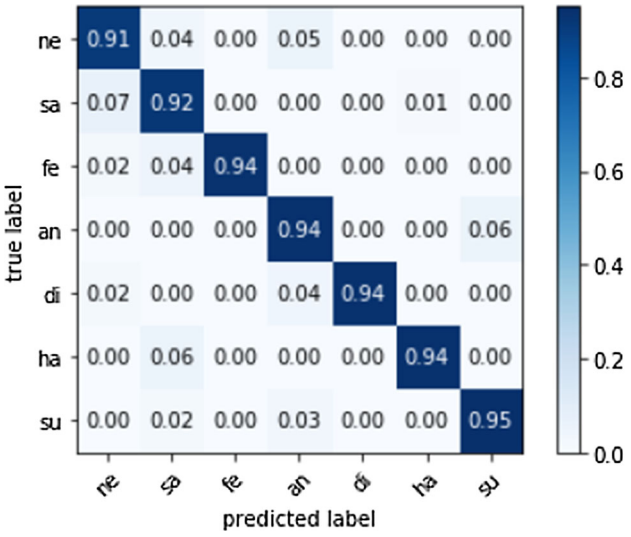

Fig. 5 Confusion matrix for AdaSVM (JAFFE dataset)

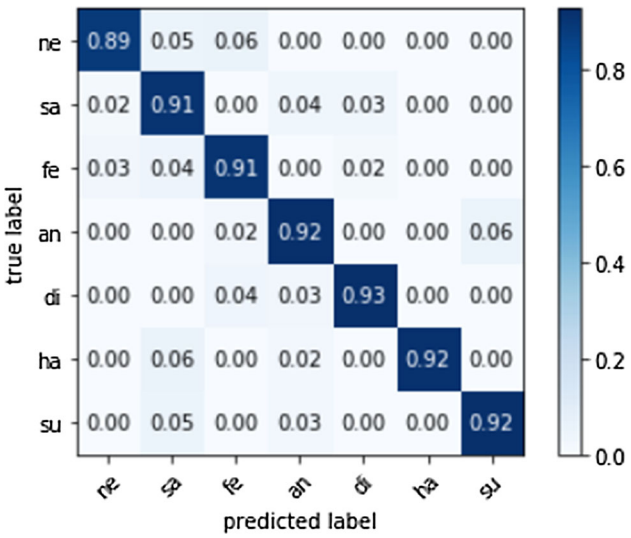

Fig. 6 Confusion matrix for AdaBoost (JAFFE dataset)

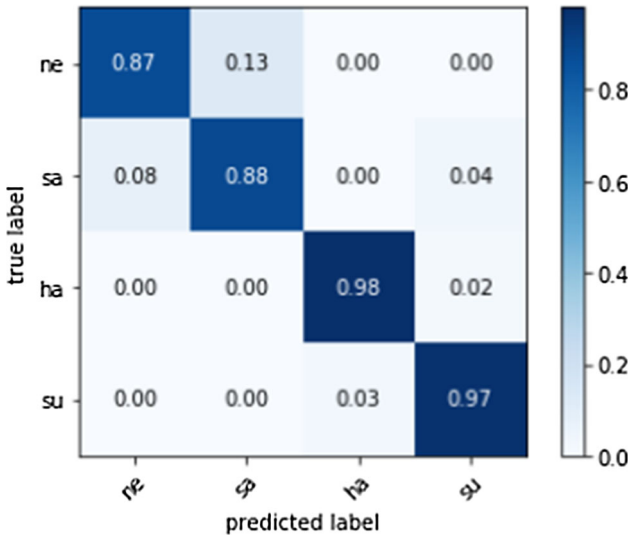

Fig. 7 Confusion matrix for Ada-AdaSVM (YALE dataset)

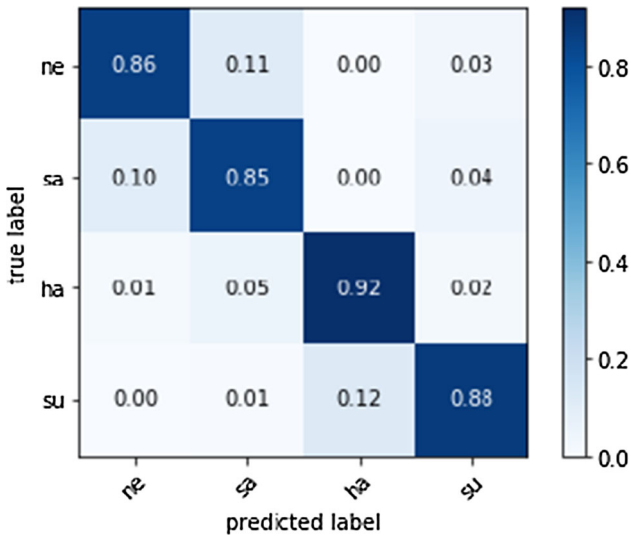

Fig. 8 Confusion matrix for SVM (YALE dataset)

for disgust, 0.97 for fear, 1.00 for happiness, 0.96 for sad and 1.00 for surprise. Comparing these values to that of AdaSVM, SVM and AdaBoost indicate that it is a superior model (see diagonals of the various matrices of the different datasets for verification).

5.2 Accuracy performances of methods

Relatively, the model performed poorly for predicting "neutral" facial expression in the JAFFE dataset. However, this situation is similar for all the other algorithms that were tested. Again, it can be observed that Ada-AdaSVM outperformed SVM with an average of margin of 2% in all the classes (see Figs. 19, 20, 21, 22). A similar pattern was observed in all the datasets including the Yale dataset, although it has only four (4) classes. The algorithm performed better on the Jaffe dataset when compared to the others. However, with regard to the prediction of the "happy" class, the model performed better on the Yale dataset than the others.

5.3 Processing speed

The algorithm was run on an HP Pavilion Notebook with AMD A8-7410 APU Radeon R5 Graphics, (4 CPUs), ~ 2.2 GHz processor. The memory capacity of the system is 8192 MB RAM. The comparative assessment of the proposed method (Ada-AdaSVM) against others in terms of training and testing times is shown in Figs. 23 and 24, respectively.

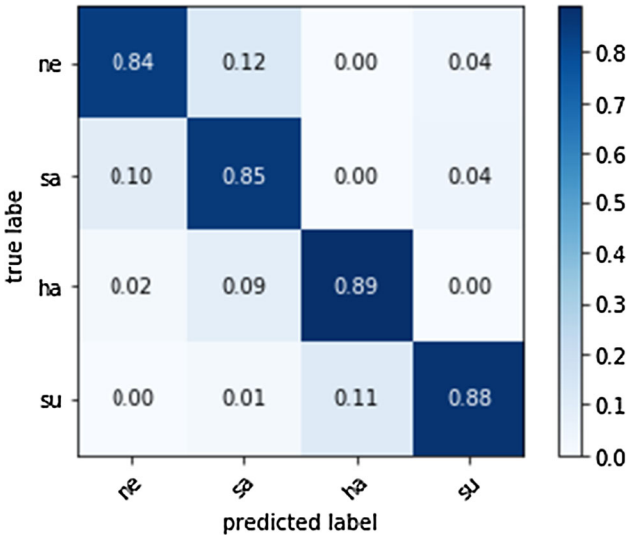

Fig. 9 Confusion matrix for AdaSVM (YALE dataset)

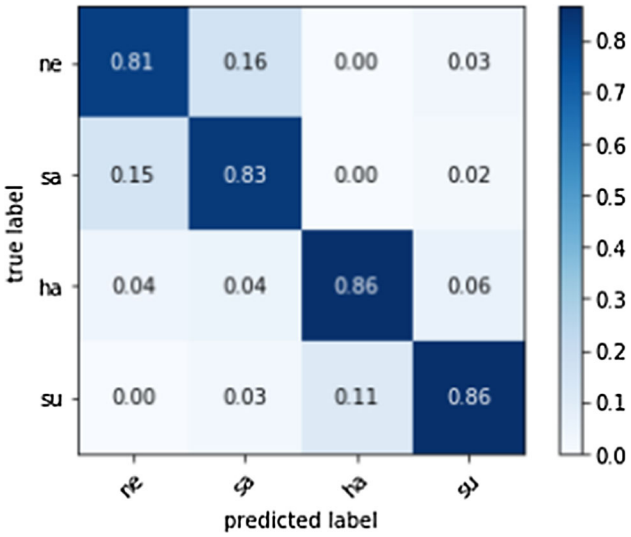

Fig. 10 Confusion matrix for AdaBoost (YALE dataset)

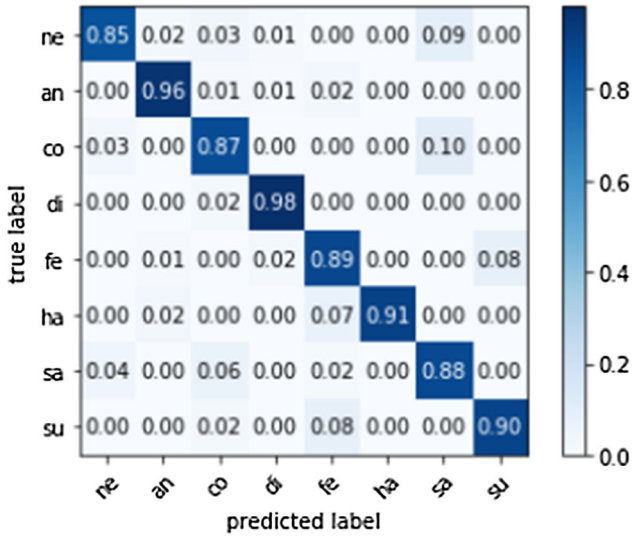

Fig. 11 Confusion matrix for Ada-AdaSVM (TFEID dataset)

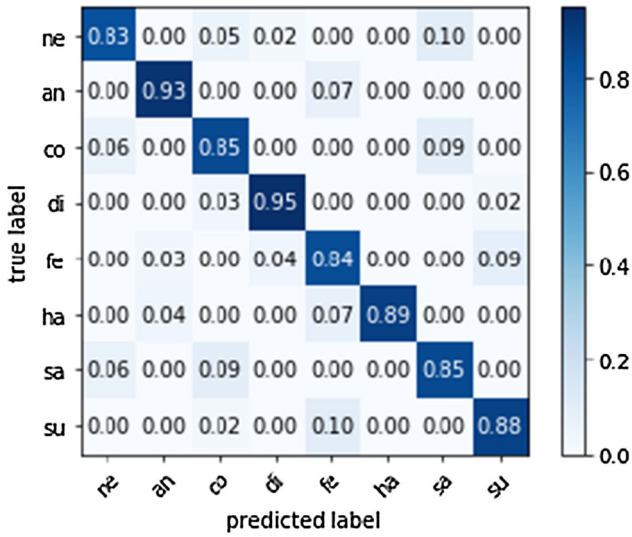

Fig. 12 Confusion Matrix for SVM (TFEID dataset)

To evaluate the speed of the model, the four datasets were combined, and Ninety-six (96) images were selected for training and 72 for testing. This included 5 spatial scales and 8 orientations, and a 96 × 96 image location was used. This produced 368,640 possible features. A fragment of the Gabor feature with the best classification performance for the current boosting distribution amounted to 576 features was selected using AdaBoost feature selection algorithm.

The combined dataset was partitioned into training and testing using a tenfold cross-validation method. The results indicated that when all the classifiers are trained with the same thresholds of Gabor features and tested with equal datasets, the performance of Ada-AdaSVM is significantly higher ($p < 0.05$). This characteristic was evident in all the dataset evaluated for both training and testing.

5.4 Statistical evidence of Ada-AdaSVM supremacy

A one-way ANOVA analysis was conducted for all the recognition accuracies to determine whether there is a statistically significant difference between the output of Ada-AdaSVM versus AdaSVM, Ada-AdaSVM versus SVM and Ada-AdaSVM versus AdaBoost in all four databases. The analysis of the variances and the corresponding Tukey simultaneous plot at a 95% confidence interval indicated that the corresponding means are

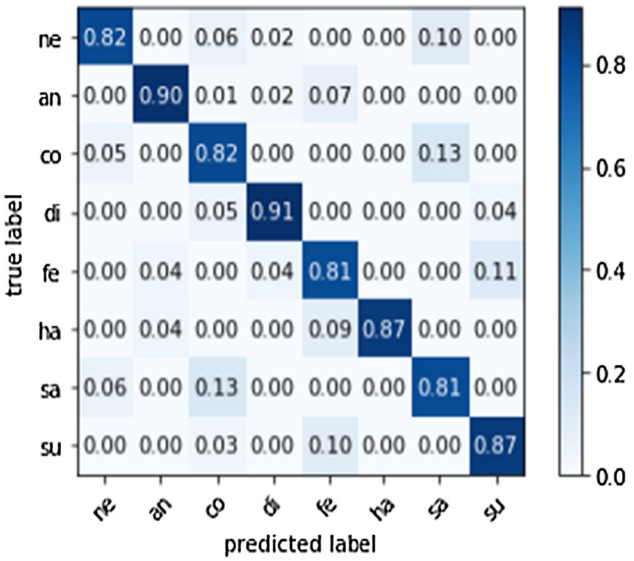

Fig. 13 Confusion matrix for AdaSVM (TFEID dataset)

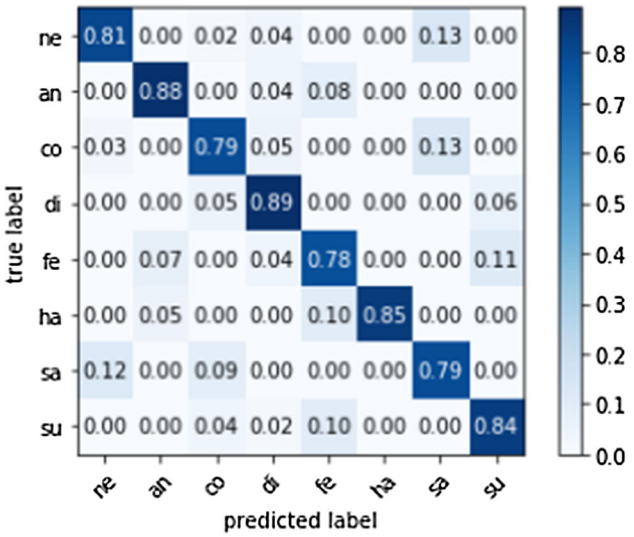

Fig. 14 Confusion matrix for AdaBoost (TFEID dataset)

significantly different ($p < 0.05$) and that Ada-AdaSVM is the best.

Again, for the testing speed, there was a statistically significant difference between groups as was shown by one-way ANOVA analysis (F (3, 12) = 4.101, p = 0.032). A Tukey post hoc test revealed that the time to complete the problem was significantly lower in Ada-AdaSVM than all the other methods, after taking the Ada-AdaSVM and SVM (p = 0.022), Ada-AdaSVM and AdaSVM (p = 0.040) and Ada-AdaSVM and AdaBoost (p = 0.037). Similarly, the training speed was significantly lower ($p < 0.05$) in Ada-AdaSVM than all the other methods.

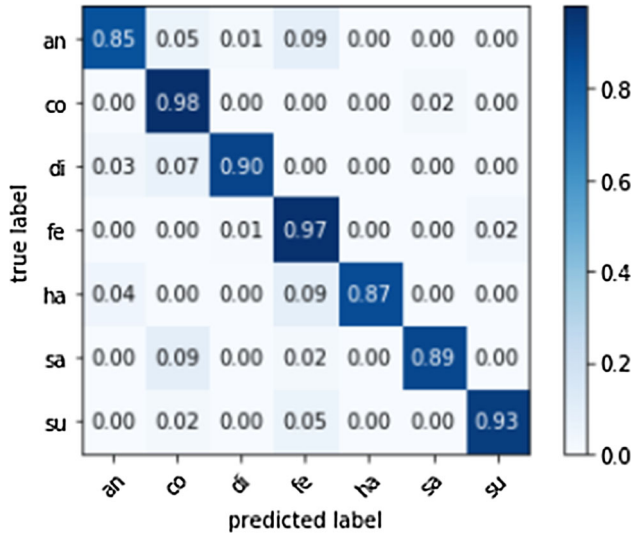

Fig. 15 Confusion matrix for Ada-AdaSVM (CK + 48 dataset)

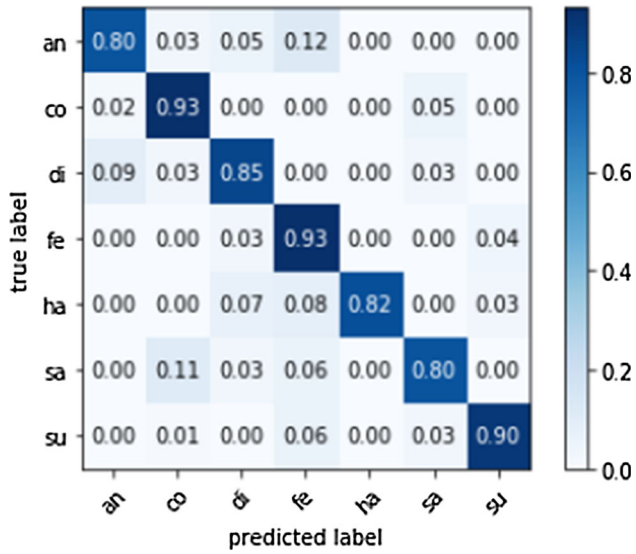

Fig. 16 Confusion matrix for SVM (CK + 48 dataset)

6 Conclusions and future work

This study proposes a novel algorithm "Ada-AdaSVM" that seeks to outperform existing facial expression classification algorithms. Specifically, it argues that by extending the original linearity of two classifiers (i.e., AdaBoost and SVM) to a multiclass problem and applying the Error-Correcting Output Codes (ECOC), one can improve prediction accuracy and speed. In this, an AdaBoost-based feature selection tool was used to select a series of uncomplicated classifiers to form a strong classifier. Then a fusion of multiclass SVM and AdaBoost hybrid classifier was used to train the model. To validate the expected improvement in prediction accuracy and speed, this method

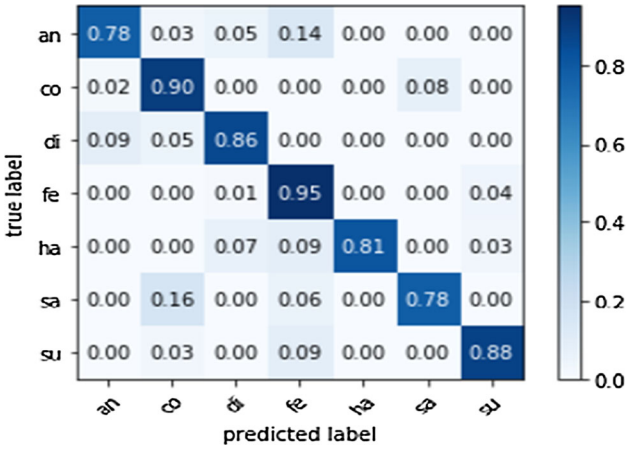

Fig. 17 Confusion matrix for AdaSVM (CK + 48 dataset)

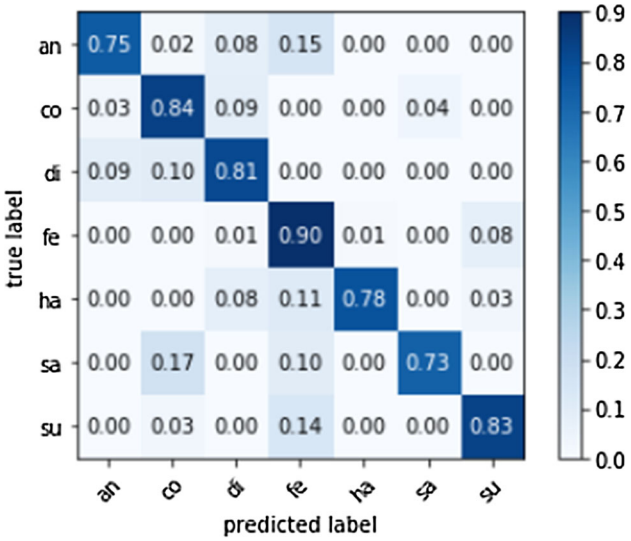

Fig. 18 Confusion matrix for AdaBoost (CK + 48 dataset)

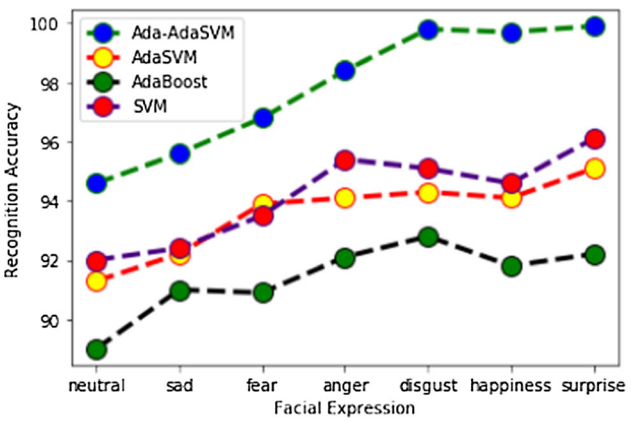

Fig. 19 Performance in Jaffe database

was evaluated by comparing it to SVM, AdaSVM and AdaBoost on the Yale, Jaffe, Taiwanese (TFEID) and

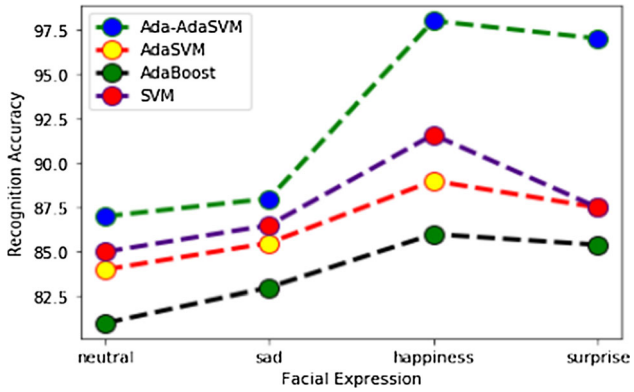

Fig. 20 Performance in Yale database

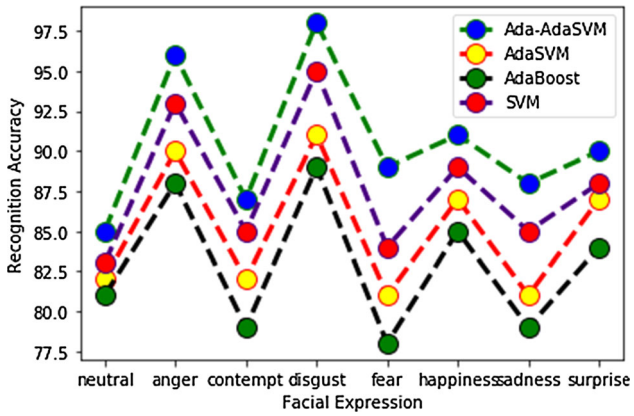

Fig. 21 Performance in TFEID

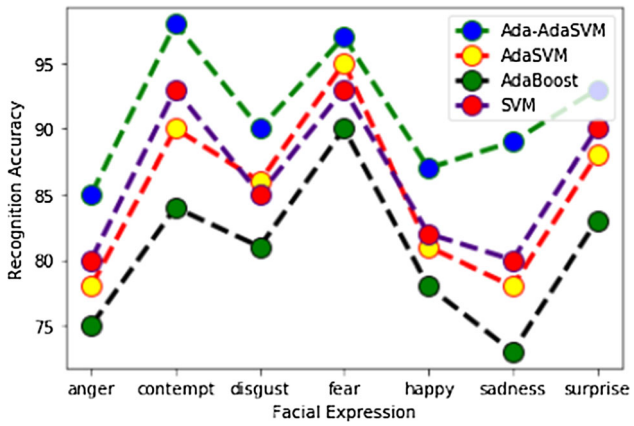

Fig. 22 Performance in CK + 48

CK + 48 facial expression datasets. The findings show that Ada-AdaSVM outperforms existing algorithms with regard to prediction accuracy and speed.

Although Ada-AdaSVM has been demonstrated to outperform other algorithms in this study, the findings cannot be generalized. This is because the study focused on facial expression classifications, and therefore, the findings may not apply to other forms of image classifications.

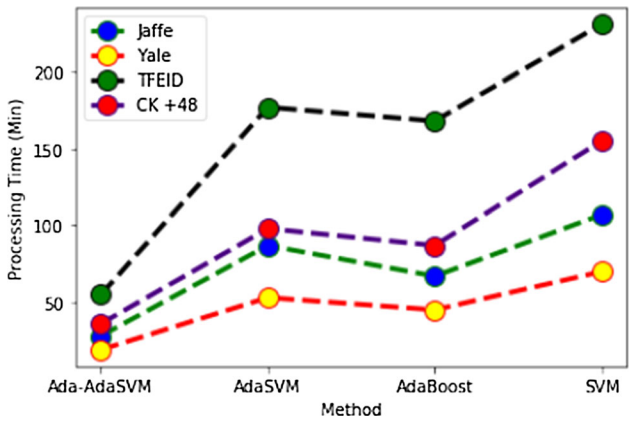

Fig. 23 Comparing the training speed of different methods

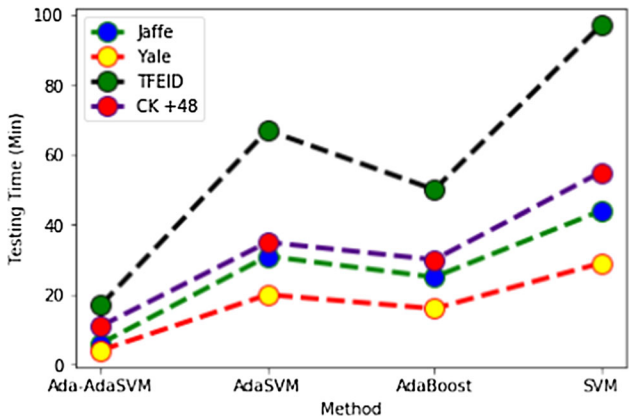

Fig. 24 Comparing testing speed of different method

Accordingly, there is a need for further studies to be conducted to validate the efficacy of the model on other types of image classification. Some advanced machine learning techniques such as the clustering of bandits may be explored as a feature selection to investigate its behavior in future work.

In addition, there is the need to evaluate how this algorithm will perform when compared to some of the most popular models for spatio-temporal data analysis including convolutional neural network (CNN) and recurrent neural network (RNN).

Declarations

Conflict of interest Conflict of interest and authorship confirmation form please check the following as appropriate: all authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version. This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue. The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript.

References

1. Mahadik K, Wu Q, Li S, Sabne A (2020) Fast distributed bandits for online recommendation systems. In: Proceedings of the 34th ACM international conference on supercomputing, pp 1–13
2. Chawla NV, Lazarevic A, Hall LO, Bowyer KW (2003) SMOTEBoost: improving prediction of the minority class in boosting. In: Lavrač N, Gamberger D, Todorovski L, Blockeel H (eds) Lecture Notes in Computer Science, vol 2838, pp 107–119. Springer Verlag, Berlin Heidelberg
3. Huang Z, Yang C, Chen X, Huang K, Xie Y (2019) Adaptive over-sampling method for classification with application to imbalanced datasets in aluminum electrolysis. In: Neural computing and applications, pp 1–17
4. Twala B, Cartwright M, Shepperd M (2006) Ensemble of missing data techniques to improve software prediction accuracy. In: Proceedings—international conference on software engineering, May 2014, pp 909–912. https://doi.org/https://doi.org/10.1145/1134285.1134449
5. Kim D (2005) Improving prediction performance of neural networks in pattern classification. Int J Comput Math 82(4):391–399. https://doi.org/10.1080/00207160420003018 06
6. Wei S, Zuo D, Song J (2012) Improving prediction accuracy of river discharge time series using a wavelet-NAR artificial neural network. J Hydroinform 14(4):974–991. https://doi.org/10.2166/hydro.2012.143
7. Błaszczyński J, Stefanowski J (2015) Neighbourhood sampling in bagging for imbalanced data. Neurocomputing 150(PB):529–542. https://doi.org/10.1016/j.neucom.2014.07.064
8. Kuncheva LI, Skurichina M, Duin RPW (2002) An experimental study on diversity for bagging and boosting with linear classifiers. Inf Fusion 3(4):245–258. https://doi.org/10.1016/S1566-2535(02)00093-3
9. Owusu E, Zhan Y, Mao QR (2014) A neural-AdaBoost based facial expression recognition system. Expert Syst Appl 41(7):3383–3390. https://doi.org/10.1016/j.eswa.2013.11.041
10. Deng L (2014) A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Trans Signal Inf Process 3(2):1–29. https://doi.org/10.1017/ATSIP.2013.99
11. Auria L, Moro RA (2008) Support vector machines (SVM) as a technique for solvency analysis (No. 811), Berlin
12. Tu JV (1996) Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. J Clin Epidemiol 49(11):1225–1231. https://doi.org/10.1016/S0895-4356(96)00002-9
13. Guo Y, Liu Y, Oerlemans A, Lao S, Wu S, Lew MS (2016) Deep learning for visual understanding: a review. Neurocomputing. https://doi.org/10.1016/j.neucom.2015.09.116
14. Mayr A, Binder H, Gefeller O, Schmid M (2014) The evolution of boosting algorithms: from machine learning to statistical modelling. Methods Inf Med 53(6):419–427. https://doi.org/10.3414/ME13-01-0122
15. Schapire RE (2013) Explaining adaboost. Empirical Inference: Festschrift in Honor of Vladimir N Vapnik. https://doi.org/10.1007/978-3-642-41136-6_5
16. Wang Y, Ai H, Wu B, Huang C (2004) Real time facial expression recognition with adaboost. Proc Int Conf Pattern Recognit 3:926–929. https://doi.org/10.1109/ICPR.2004.1334680
17. Narsky I, Porter FC (2013) Reducing multiclass to binary. Stat Anal Tech Part Phys 1:371–379. https://doi.org/10.1002/9783527677320.ch16
18. Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). Ann Stat 28(2):337–407. https://doi.org/10.1214/aos/1016218223

19. Lin C-J, Hsu C-W (2002) A comparison of methods for multi-class support vector machine. IEEE Trans Neural Netw 13(2):415–425
20. Senior A (1998) A hidden Markov model fingerprint classifier. In: Signals, system and computers, pp 306–310
21. Amami R, Ben Ayed D, Ellouze N (2013) The challenges of SVM optimization using Adaboost on a phoneme recognition problem. In: 4th IEEE international conference on cognitive infocommunications, CogInfoCom 2013—proceedings, pp 463–468. https://doi.org/https://doi.org/10.1109/CogInfoCom.2013.6719292
22. Li X, Wang L, Sung E (2008) AdaBoost with SVM-based component classifiers. Eng Appl Artif Intell 21(5):785–795. https://doi.org/10.1016/j.engappai.2007.07.001
23. Satiyan M, Nagarajan R (2010) Recognition of facial expression using Haar-like feature extraction method. In: 2010 international conference on intelligent and advanced systems, ICIAS 2010, pp 1–4. https://doi.org/https://doi.org/10.1109/ICIAS.2010.5716228
24. Owusu E, Zhan YZ, Mao QR (2014) An SVM-AdaBoost-based face detection system. J Exp Theor Artif Intell 26(4):477–491. https://doi.org/10.1080/0952813X.2014.886300

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.