# Application based on Hybrid CNN-SVM and PCA-SVM Approaches for Classification of Cocoa Beans

AYIKPA Kacoutchy Jean[1], MAMADOU Diarra[2], BALLO Abou Bakary[3], GOUTON Pierre[4], ADOU Kablan Jérôme[5]
ImVia, Université Bourgogne Franche-Comté, Dijon, FRANCE[1, 4]
LaMI, Université Felix Houphouët-Boigny, Abidjan, CÔTE D'IVOIRE[1, 2, 3, 5]
UREN, Université Virtuelle de Côte d'ivoire, Abidjan, CÔTE D'IVOIRE[1]

***Abstract*—In our study, we propose a hybrid Convolutional Neural Network with Support Vector Machine (CNN-SVM) and Principal Component Analysis with support vector machine (PCA-SVM) methods for the classification of cocoa beans obtained by the fermentation of beans collected from cocoa pods after harvest. We also use a convolutional neural network (CNN) and support vector machine (SVM) for the classification operation. In the case of the hybrid model, we use a convolutional network as a feature extractor and the SVM is used to perform the classification operation. The use of PCA-SVM allowed for a reduction in image size while maintaining the main features still using the SVM classifier. Radial, linear and polynomial basis function kernels were used with various control parameters for the SVM, and optimizers such as the Stochastic Gradient Descent (SGD) algorithm, Adam, and RMSprop were used for the CNN softmax classifier. The results showed the robustness of the hybrid CNN-SVM model which obtained the best score with a value of 98.32% then the PCA-SVM based model had a score of 97.65% outperforming the standard CNN and SVM classification algorithms. Metrics such as accuracy, recall, F1 score, mean squared error (MSE), and MCC have allowed us to consolidate the results obtained from our different experiments.**

***Keywords—Support vector machine; convolutional neural network; cocoa beans; principal component analysis; hybrid method***

## I. Introduction

Côte d'Ivoire is the world's largest producer of cocoa [1] and cocoa is an important cash crop in the world. The cocoa culture produces beans from the ripe pod seeds of the Theobroma plant [2]. Cocoa beans are the main raw material for chocolate [3] and the first step in this process is fermentation.

Fermentation is an essential step in cocoa processing, and it has an impact on the flavor, color, and aroma of cocoa products [4]. Unfermented cocoa beans do not have the full flavor of chocolate, but fermentation triggers chemical changes within the cocoa bean that contribute to the development of chocolate flavor [5]. Once harvested, farmers open the cocoa pods, extract the cocoa seeds with the pulp and fill wooden boxes or containers to begin fermentation [6]. Using quality dried cocoa beans from the fermentation process allows for obtaining better-finished products. The process of detecting the quality of dried cocoa beans is a tedious task and requires special attention, hence the need to use computer vision that will allow an image to specify its category of it. In recent years, computer vision has an important role in agricultural production with the use of machine learning and more specifically deep learning convolutional neural networks [7]. We can note here the popular image analysis techniques in machine learning such as Support vector machine (SVM), Artificial Neural Network (ANN), Convolutional Neural Networks (CNN), Normalized Difference Vegetation Index (NDVI), and statistical tools such as correlation and regression analysis, etc [8]. In order to facilitate the classification of cocoa beans, we proposed the methods of CNN, SVM, hybrid CNN-SVM, and principal component analysis a with support vector machine (PCA-SVM). The general principle of the hybrid model is to automatically extract features based on the CNN and do the classification using the SVM classifier, while the PCA-SVM reduces the image size while keeping the main features still using the SVM classifier. All these methods were used to detect the category of cocoa dried seeds and then we compared them to come out with the best method. Our technique can evolve into an industrial application with an appropriate integration framework, replacing the traditional method of quality control of cocoa beans. Thus, our study can be integrated into a computer vision system and implemented in the cocoa production and processing chain, resulting in a state-of-the-art automatic solution. The proposed approach could benefit the industry by enabling them to accurately determine the quality of cocoa beans.

This paper is organized as follows: Section II presents some previous work, and Section III details the methodology and the material we propose. Section IV presents the results obtained and discusses them, and we conclude in Section V.

## II. Related Work

Several works focused in this area and we can cite some of them, namely, Oliviera et al. used handcrafted features calculated from the beans provided by image analysis tools, and then the random decision forest predictor was used to classify the samples. This experiment yielded an accuracy of 92% [9]; Kaghi et al. used a pre-trained AlexNet CNN as a generic feature extractor of a 2D image whose dimensions were reduced using PCA + TSNE and finally classified using a simple machine learning algorithm like KNN, and Naïve Bayes Classifier. These results could match a CNN Softmax classifier [10]; Barbon et al. used machine learning-based methods namely J48, Naïve Bayes, K-NN, Random Forest, SVM, MLP, and Fuzzy approaches to predict the storage time of pork, and these methods provided the accuracies which ranged from 78.26 to 94.41% [11]. A. and Renjith provide a special architecture to identify the features of different classes of Durian fruit, the image processing model allows the classification to be divided into two parts: feature extraction and classification of the fruit used. The use of edge detection and color extraction provide correct feature extraction of durian, the performance is measured using non-destructive machine learning techniques such as SVM, GNB, and Random Forest. The results obtained provide the best accuracy of 89.3% using the SVM technique and 84.3% using Random Forest [12]; Harel et al. proposed maturity classification algorithms. Their algorithms were applied to the maturity classification of peppers. Maturity classification achieved 98.2% and 97.3% accuracy for two-class classification between mature and immature classes of red and yellow peppers, respectively, and 89.5% and 97.3% accuracy for four-class maturity classification. The random forest algorithm has been shown to be very robust [13]. Elleuch et al. explored a method focusing on the use of two classifiers in this case Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for offline Arabic handwriting recognition, the performance of their methods was compared with the character recognition accuracies obtained from the state of the art of optical Arabic character recognition, producing favorable results [14].

## III. Material and Method

### A. Materials

The digital images of cocoa beans used for the study were obtained at YAKASSE 1 (longitude: -3.77374 latitude: 5.23841) village located near GRAND-BASSAM in Côte d'Ivoire. These cocoa bean samples were classified as follows:

- Category 1 beans: fermented and dried cocoa beans of superior quality.
- Category 2 beans: fermented and dried cocoa beans of intermediate quality.
- Category 3 beans: non-fermented and dried cocoa beans.

Once the data is obtained as shown in Fig. 1, we will proceed to the pre-processing that will extract the seeds from each image.

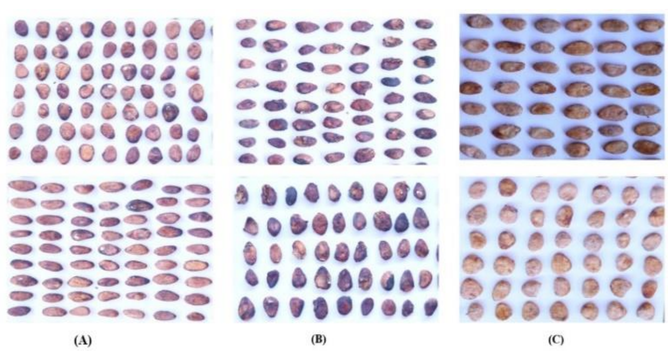



Fig. 1. Cocoa Beans Sample Images from the Three Classes: (A) Category 1 Beans; (B) Category 2 Beans; (C) Category 3 Beans.

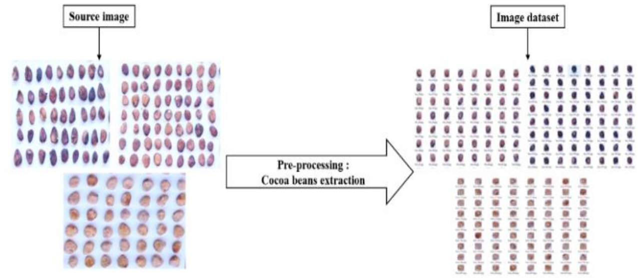



Fig. 2. General Schema of the Dataset.

We describe the preprocessing stage now

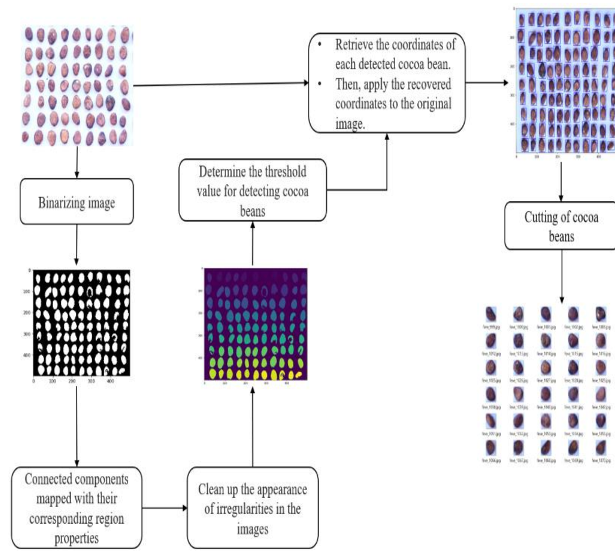



Fig. 3. Preprocessing Cocoa Beans Extraction.

After the preprocessing step as shown in Fig. 2 and Fig. 3, we obtained a dataset of 3470 images of cocoa beans, including 917 images of beans of category 1, 1675 images of beans of category 2, and 878 images of beans of category 3. We split the dataset into 60% for training, 20% for testing, and 20% for validation.

The table I presents the data split description.

TABLE I. Description of the Data Split

| ***Dataset*** | ***Training set*** | ***Validation set*** | ***Test set*** |
|---|---|---|---|
| 100% | 60% | 20% | 20% |
| 3470 | 2082 | 694 | 694 |

We trained the models on a Windows 10 system with an Intel(R) Core™ i7-8650U processor, 16 GB of random-access memory (RAM), and an NVIDIA GeForce MX150 graphics processing unit (GPU). The models are configured in Python using the Keras version 2.4 API with the TensorFlow version 2.4 backend and CUDA/CuDNN dependencies for GPU acceleration.

### B. Methods

Our method is segmented into three main parts as shown in Fig. 4, namely preprocessing as seen in the materials section, feature extraction, and classification.

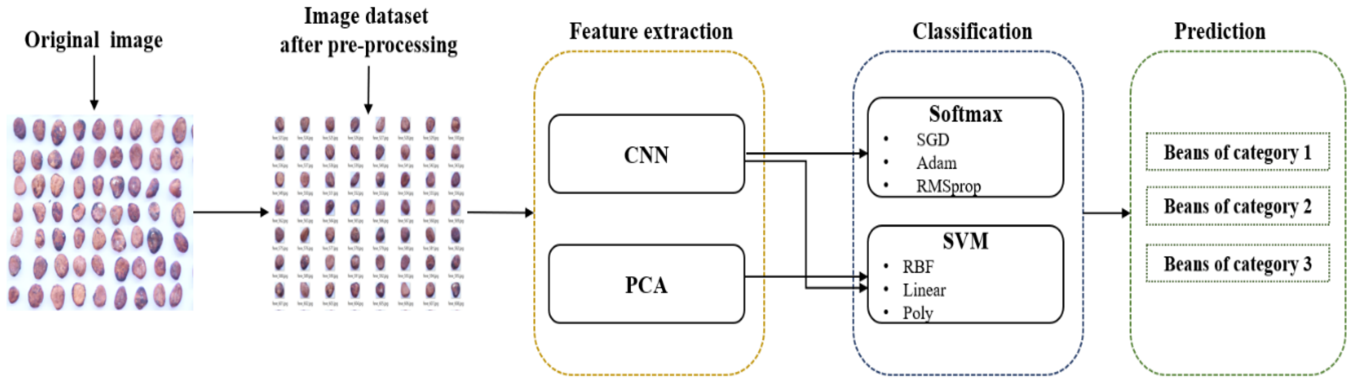



Fig. 4. Architecture of the Model for the Classification of Cocoa Beans.

*1) Feature extraction:* We proposed two feature extractors namely the CNN and the PCA.

*a) CNN:* The CNN we used in our study consists of two convolutional layers with 32 filters each and a 3x3 size kernel with a Relu activation function. The convolutional layer allows for the generation of a particular feature map by applying a filter that examines the whole image. Each of the convolutional layers has a max-pooling layer of 2x2 core, it is a subsampling layer. The subsampling of the pooling layer consists of extracting the most important value of each pattern from the feature map. This layer reduces the parameters and computations in the network, this layer improves the performance of the network and avoids overlearning [15]. Finally, a Flatten layer that flattens the feature map and reduces its size. The Table II gives the description of CNN.

*b) PCA:* Principal component analysis (PCA) is a technique that is applied in applications such as dimensionality reduction, data compression, feature extraction, and data visualization. PCA allows a set of correlated variables X to be transformed into a smaller number y with y<X of uncorrelated variables called principal components while retaining as much variability of the original data. One of the features of PCA is image compression a technique that reduces the size of an image while retaining as much of the image quality as possible [16].

*2) Classification :* Classification is a task that uses machine learning algorithms that learn to assign a class label to examples in a domain for a given problem. There are many types of classification tasks in machine learning and specialized modeling approaches that can be used for each [17]. In our study, we used the Softmax and SVM classifiers.

*a) Classifier Softmax:* The Softmax classifier is a generalization of the binary form of logistic regression, It has been used in deep learning more precisely in the field of computer vision to classify the vectors obtained after feature extraction [18]. In its operation, the mapping function F is defined such that it takes a set of input data x and maps to output class labels from a simple dot product of the data x and the weight matrix W.

$$F(x_i, W) = W * x_i \quad (1)$$

The Softmax score function gives a probability based on the final score. The sum of the probability of all categories is equal to one [18]. The equation is as follows:

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \quad (2)$$

The Softmax loss function can be viewed as the entropy of two probabilities, as shown in the following equation:

$$H(p,q) = -\sum_x p(x) \log q(x) \quad (3)$$

We will use optimizers which are algorithms used to minimize the loss function. These functions are:

- SGD which stands for Stochastic Gradient Descent, is a gradient descent optimizer that is used in machine learning and deep learning. Stochastic means a system that is connected or linked with a random probability [19].
- Adam is the extended version of stochastic gradient descent that could be implemented in various deep learning applications such as computer vision and natural language processing [19].
- RMSprop which stands for Root Mean Squared Propagation is an extension of gradient descent and the AdaGrad version of gradient descent that uses a decreasing average of partial gradients to tailor the size of each parameter step. The use of a decreasing moving average allows the algorithm to forget about bad gradients and focus on the best gradients observed during the search progress, overcoming the limitations of AdaGrad [19].

TABLE II. CNN ARCHITECTURE

| *Layer* | *Output Shape* | *Parameter Size* |
|---|---|---|
| Convolutional 1 | (58, 58, 32) | 896 |
| Pooling | (29, 29, 32) | 0 |
| Convolutional 2 | (27, 27, 32) | 9248 |
| Pooling | (13, 13, 32) | 0 |
| Flattening | 5408 | 0 |
| Total parameter | 703 012 | |
| Trainable parameter | 703 012 | |
| Non-Trainable parameter | 0 | |

*b) Classifier SVM:* The support vector machine (SVM) is a supervised algorithm used in machine learning. It tries to find a hyperplane that best separates the different data. The SVM is based on statistical approaches, it allows the classification of the data and assigns to each data a specific score as a basis for evaluation. SVM can be used for both regression and classification tasks. However, it is more commonly used for classification objectives [20]. The SVM constructs a hyperplane or a set of hyperplanes in a high or infinite dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyper-plane that has the largest distance to the nearest training data points of any class, because in general the larger the margin, the smaller the generalization error of the classifier [21]. The SVM solves the following equation:

$$\min_{\omega,b,\zeta} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{m}\zeta_i$$

$$Subject\ to\ yi(\omega^T\,\Phi(x_i) + b) \geq 1 - \zeta_i,$$

$$\zeta_i \geq 0, i = 1, \dots, n \tag{4}$$

The loss function is represented by the false predictions of the score function. In SVM, Multiclass SVM Loss is used. The main idea of Multiclass SVM Loss is to determine the scores given by Score Function, requiring the final score to be at least one unit higher than the incorrect score [22]. The loss function is defined by the following equation:

$$L_i = \sum_{j \neq y_i} \max(0, w_j^T x_i - w_{yi}^T x_i + \Delta) \tag{5}$$

We use the following SVM kernels in our study:

- The linear kernel and its equation is:

$$K(x, y) = x * y \tag{6}$$

- The polynomial kernel and its equation is :

$$K(x, y) = [(x \times y) + 1]\ d \tag{7}$$

- The RBF (Radial Basis Function) kernel and its equation is:

$$K(x, y) = \exp(-\gamma \|x - y\|^2). \tag{8}$$

Also, we have used the values 1 and 100 for the parameter C which is common to all SVM kernels, it allows us to correct the errors in the classification of the training examples by the simplicity of the decision surface.

### C. Evaluation Metrics

To validate the performance of the pre-trained models in our study we will use the following metrics:

- Accuracy is a performance measure that shows how well the system has classified the data into the correct class.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{9}$$

- Precision is the ratio of correctly classified positive images to the total number of true positive images.

$$Precison = \frac{TP}{TP+FP} \tag{10}$$

- Recall is the ability of a classifier to determine actual positive outcomes

$$Recall = \frac{TP}{TP+FN} \tag{11}$$

- The F1 score is the weighted average of precision and recall

$$F1\ score = 2 * \frac{Precision*Recall}{Precision+Recall} \tag{12}$$

- The Matthews correlation coefficient (MCC) is used in machine learning as a measure of the quality of classifications

$$MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{13}$$

- The mean square error of an estimator measures the average of the squared errors, i.e. the mean square difference between the estimated values and the true value.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Yi - \hat{Y}i)^2 \tag{14}$$

With TP: True positive, TN: True negative, FP: False positive, and FN: False negative.

## IV. Results

### A. CNN with Softmax

In the case of the CNN with Softmax, we can record the results presented in the Table III which takes into account the different optimizers mentioned above.

TABLE III. THE GENERAL PERFORMANCE OF THE SOFTMAX CLASSIFIER

| | ***Accuracy*** | ***Loss*** | ***Precision*** | ***F1 score*** | ***Recall*** | ***MCC*** | ***MSE*** |
|---|---|---|---|---|---|---|---|
| ***SGD*** | 92,95 | 17,93 | 93,37 | 92,95 | 92,95 | 89,2 | 3,41 |
| ***Adam*** | **95,64** | **17,18** | **95,65** | **95,64** | **95,63** | **93,18** | **2,37** |
| ***RMSprop*** | 94,63 | 36,99 | 94,83 | 94,62 | 94,63 | 91,73 | 3,18 |

The results presented in Table III show that the Adam optimizer obtains the best performance with a score of 95.64%, followed by the RMSprop and finally the SGD. We see in this experiment that the softmax classifier using the Adam optimizer is more optimal.

Fig. 5 presents the confusion matrix of each softmax experiment case; also, a histogram of the metrics has been created.

### B. SVM

In the case of SVM we can record the results presented by the Table IV which takes into account the different kernels and parameters.

The results presented in Table IV show that the SVM with the RBF kernel; and the C parameter at 100 obtained the best performance with a score of 97.32%.

We now present the confusion matrices and histograms of the metrics in Fig. 6.

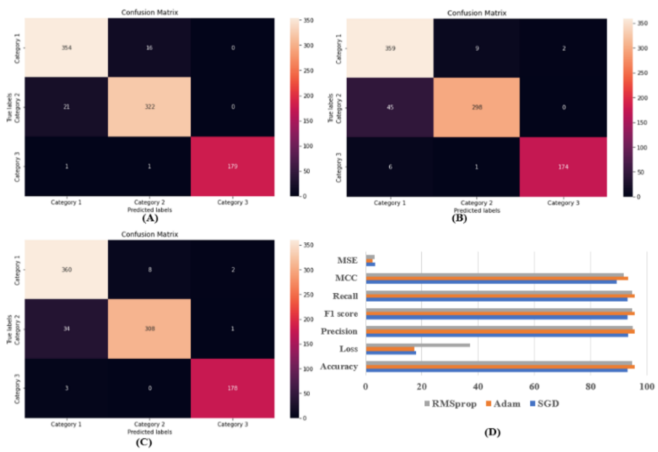
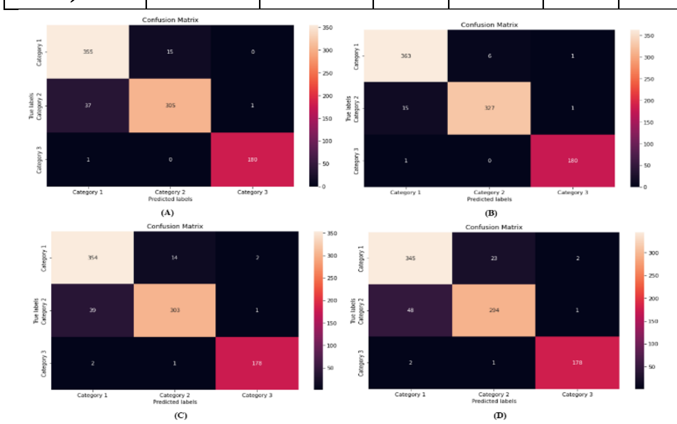


Fig. 5. (A) Confusion Matrix of Adam ; (B) Confusion Matrix of SGD ; (C) Confusion Matrix of RMSprop ; (D) Graphical Representation of Optimizer Metrics.

TABLE IV. GENERAL PERFORMANCE OF THE SVM CLASSIFIER

| | *Accuracy* | *Precision* | *F1 score* | *Recall* | *MCC* | *MSE* |
|---|---|---|---|---|---|---|
| ***SVM (rbf ; C=1)*** | 93,96 | 94,08 | 93,94 | 93,95 | 90,65 | 6,37 |
| ***SVM (rbf ; C=100)*** | **97,32** | **97,33** | **97,31** | **97,31** | **95,82** | **3,35** |
| ***SVM (linear; C=1)*** | 93,4 | 93,55 | 93,38 | 93,4 | 89,79 | 7,94 |
| ***SVM (linear; C=100)*** | 91,39 | 91,52 | 91,36 | 91,38 | 86,64 | 9,95 |
| ***SVM (poly; C=1)*** | 95,75 | 95,82 | 95,74 | 95,74 | 93,41 | 5,25 |
| ***SVM (poly; C=100)*** | 95,97 | 96,02 | 95,97 | 95,97 | 93,73 | 5,03 |



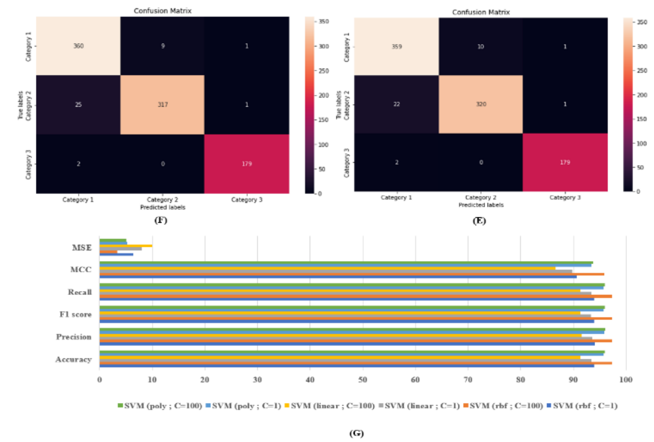


Fig. 6. (A) Confusion Matrix of SVM (RBF ; C=1) ; (B) Confusion Matrix of SVM (RBF ; C=100) ; (C) Confusion Matrix of SVM (Linear ; C=1) ; (D) Confusion Matrix of SVM (Linear ; C=100) ; (E) Confusion Matrix of SVM (Poly ; C=1) ; (F) Confusion Matrix of SVM (Poly; C=100); (G) Graphical Representation of Optimizer Metrics.

## C. CNN with SVM

For the hybrid CNN-SVM method, we can record the results presented in Table V which takes into account the different kernels and SVM parameters.

TABLE V. GENERAL PERFORMANCE OF THE CNN WITH SVM

| | *Accuracy* | *Precision* | *F1 score* | *Recall* | *MCC* | *MSE* |
|---|---|---|---|---|---|---|
| ***CNN - SVM (rbf ; C=1)*** | 95,08 | 95,29 | 95,06 | 95,07 | 92,44 | 4,92 |
| ***CNN - SVM (rbf ; C=100)*** | **98,32** | **98,34** | **98,32** | **98,32** | **97,39** | **1,67** |
| ***CNN - SVM (linear; C=1)*** | 95,86 | 95,95 | 95,85 | 95,86 | 93,59 | 4,47 |
| ***CNN - SVM (linear; C=100)*** | 95,86 | 95,95 | 95,85 | 95,86 | 93,59 | 4,47 |
| ***CNN - SVM (poly; C=1)*** | 95,41 | 95,61 | 95,4 | 95,41 | 92,96 | 5,25 |
| ***CNN - SVM (poly; C=100)*** | 98,10 | 98,11 | 98,09 | 98,09 | 97,04 | 1,90 |

The results presented in Table V show that the CNN-SVM with the RBF kernel; and the C parameter at 100 obtained the best performance with a score of 98.32%.

We now present the confusion matrices and histograms of the metrics in Fig. 7.

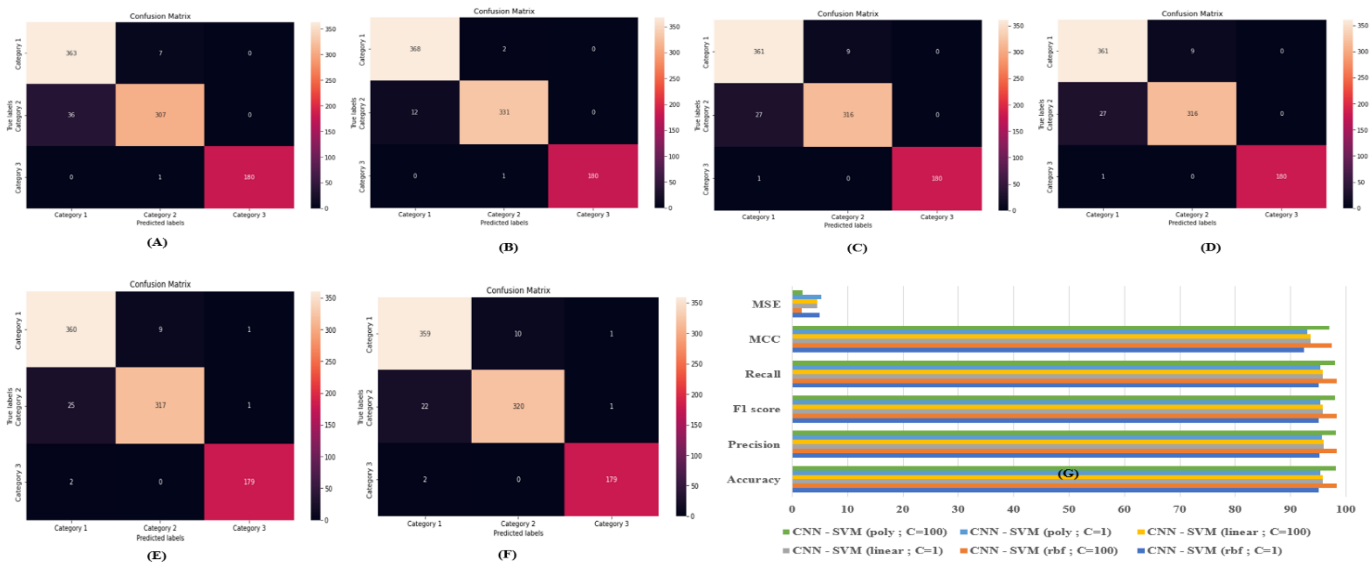



Fig. 7. (A) Confusion Matrix of CNN - SVM (rbf ; C=1) ; (B) Confusion Matrix of CNN - SVM (rbf ; C=100) ; (C) Confusion Matrix of CNN - SVM (Linear ; C=1) ; (D) Confusion Matrix of CNN - SVM (Linear ; C=100) ; (E) Confusion Matrix of CNN - SVM (Poly ; C=1).

## D. PCA with SVM

For the PCA-SVM method, we can record the results presented in Table VI which takes into account the different kernels and parameters of the SVM.

The results presented in Table VI show that the SVM with the RBF kernel; and the C parameter at 100 obtained the best performance with a score of 97.65%.

We now present the confusion matrices and histograms of the metrics in Fig. 8.

## E. Comparison of the Results of the Different Methods

We will compare the results of the different methods used in our study, namely: CNN, CNN-SVM, SVM, and PCA-SVM. The results will be represented in Table VII.

Table VII compares the best results obtained in our different experiments. It appears that the hybrid CNN-SVM method obtained the best score followed by the PCA-SVM. Fig. 9 shows the histogram of the comparison

TABLE VI. GENERAL PERFORMANCE OF THE PCA WITH SVM

| | *Accuracy* | *Precision* | *F1 score* | *Recall* | *MCC* | *MSE* |
|---|---|---|---|---|---|---|
| ***PCA - SVM (rbf ; C=1)*** | 95,41 | 95,49 | 95,4 | 95,41 | 92,89 | 5,21 |
| ***PCA - SVM (RBF; C=100)*** | **97,65** | **97,67** | **97,64** | **97,65** | **96,34** | **3,02** |
| ***PCA - SVM (linear; C=1)*** | 93,40 | 93,55 | 93,38 | 93,4 | 89,79 | 7,94 |
| ***PCA - SVM (linear; C=100)*** | 91,39 | 91,52 | 91,36 | 91,38 | 86,64 | 9,95 |
| ***PCA - SVM (poly; C=1)*** | 92,73 | 93,13 | 92,74 | 92,72 | 88,81 | 10,96 |
| ***PCA - SVM (poly; C=100)*** | 97,20 | 97,2 | 97,19 | 97,2 | 95,64 | 3,13 |

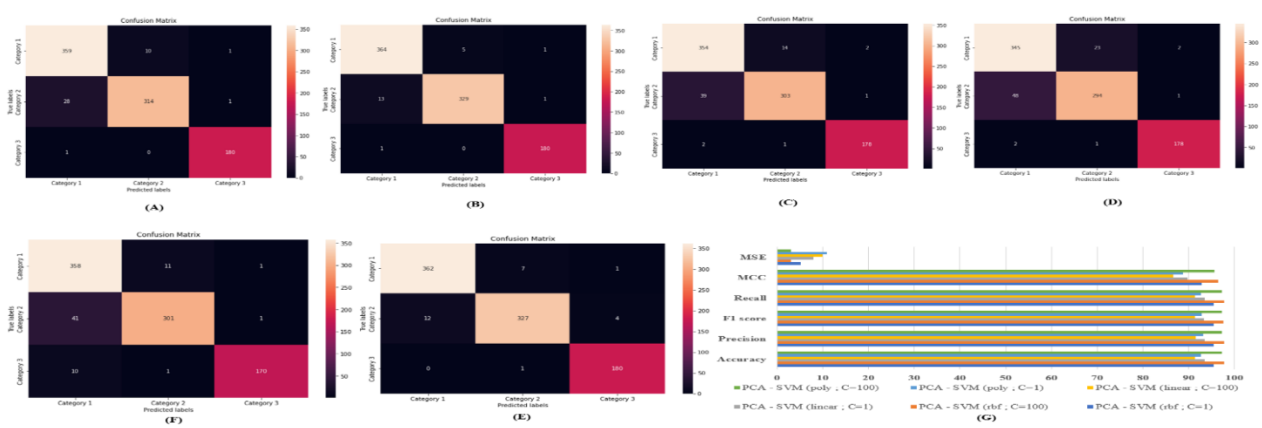



Fig. 8. (A) Confusion Matrix of PCA - SVM (rbf ; C=1) ; (B) Confusion Matrix of PCA - SVM (rbf ; C=100) ; (C) Confusion Matrix of PCA - SVM (Linear ; C=1) ; (D) Confusion Matrix of PCA - SVM (Linear ; C=100) ; (E) Confusion Matrix of PCA - SVM (Poly ; C=1) ; (F) Confusion Matrix of PCA - SVM (Poly ; C=100); (G) Graphical Representation of Optimizer Metrics.

TABLE VII. COMPARATIVE TABLE OF THE BEST RESULTS OF OUR EXPERIMENTS

| ***Models*** | ***Accuracy*** |
|---|---|
| **CNN - SVM (RBF; C=100)** | **98,32** |
| PCA - SVM (RBF; C=100) | 97,65 |
| SVM (RBF; C=100) | 97,32 |
| Softmax with Adam optimizer | 95,64 |

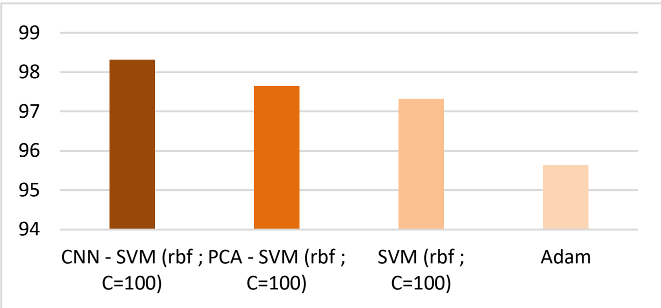



Fig. 9. Histogram of Best Scores of Experiments.

### *F. State-of-the-art Comparison*

The results of our experiments have given better results than the state of the art and Table VIII presents the results. These results are also represented by the histogram as shown in Fig. 10.

TABLE VIII. COMPARISON OF THE RESULTS OF OUR EXPERIMENTS WITH THE STATE–OF-THE-ART

| ***Models*** | ***Accuracy*** |
|---|---|
| Oliviera et al. [9] | 92 |
| CNN - SVM (rbf ; C=100) | **98,32** |
| PCA - SVM (rbf ; C=100) | **97,65** |
| SVM (rbf ; C=100) | **97,32** |
| Softmax with Adam optimizer | **95,64** |

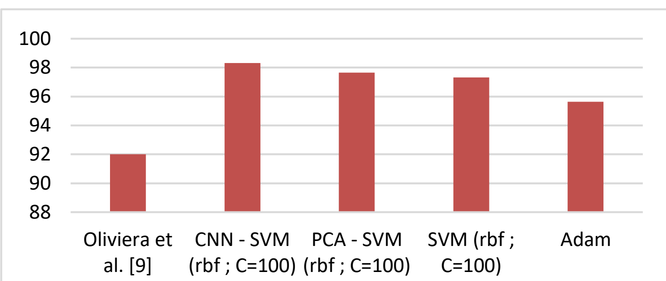



Fig. 10. Histogram of the Results of our Experiments with the State-of-the-Art.

## V. DISCUSSION

The analysis of digital images of a cocoa bean using CNN and PCA-based feature extractors were used to then perform the classification of cocoa beans from softmax and SVM classifier. The work resulted in the following:

First, we used the CNN coupled with the softmax classifier using several optimizers in this case Adam, RMSprop, and SGD. The Adam optimizer obtains the best performance with a score of 95.64%, followed by RMSprop and SGD. These results are presented in Table III. In the second step, we used the SVM classifier using several kernels such as rbf, linear, and poly with the parameters C with a value of 1 and 100. We thus obtained a score of 97.32% which represents the best accuracy coming from the rbf kernel with C = 100, these results are presented in Table IV. In a third step, we used the hybrid CNN-SVM method with the same parameters used by the SVM, again we have a score of 98.32% achieved with the rbf kernel and C=100, these results are presented in Table V. Finally in a fourth time we have the PCA-SVM method which also uses the same parameters of the SVM and obtained a score 97.65% with the kernel rbf and C=100, these results are presented in Table VI. At the end of our work, we realize that the hybrid CNN-SVM method obtained the best accuracy of all the methods used as shown in Table VII of the comparative study, and also of the methods of previous studies in the literature review as shown in Table VIII.

## VI. CONCLUSION

Cocoa production is an area of research that needs the use of automated methods for product quality assessment. The results obtained showed that feature extractions based on CNN coupled with an SVM classifier are promising systems to classify cocoa beans according to quality. Also, we used principal component analysis to reduce the size of our data while designing the main features and this allowed us to have a satisfactory result, which results in showing that we can minimize the computational time of the classifiers proceeding to a reduction of the dimensions. The result of the hybrid CNN-SVM method obtained the best score of all the methods used including the one in the literature. We achieved our goal because the hybrid method gives us a better score. In future work, we will be able to use texture extraction methods and pre-trained CNN sets for more accuracy. Also, a study of the quality of cocoa beans based on the approach of the maturity of cocoa pods will initially distinguish the best pods. The harvest of unripe pods gives beans of poor quality, while a pod too ripe has beans that begin to germinate or alter inside the pod. The classification of cocoa beans according to their shape or morphological parameters will also help to obtain quality beans.

## ACKNOWLEDGMENT

Our gratitude goes to the village community of YAKASSE, located in the town of GRAND-BASSAM in CÔTE D'IVOIRE, who kindly put their plantations at our disposal for our studies.

## REFERENCES

[1] « Fèves de cacao : production mondiale par pays 2022 », Statista. https://fr.statista.com/statistiques/565101/production-mondiale-feves-cacao-volume-par-pays/ (consulté le 30 juillet 2022).

[2] Nair, K. P. (2021). Cocoa (Theobroma cacao L.). In Tree Crops (pp. 153-213). Springer, Cham.

[3] Urbańska, B., & Kowalska, J. (2019). Comparison of the total polyphenol content and antioxidant activity of chocolate obtained from roasted and unroasted cocoa beans from different regions of the world. Antioxidants, 8(8), 283.

[4] L. Samagaci, H. Ouattara, S. Niamké, et M. Lemaire, « Pichia kudrazevii and Candida nitrativorans are the most well-adapted and relevant yeast species fermenting cocoa in Agneby-Tiassa, a local Ivorian cocoa producing region », Food Res. Int., vol. 89, p. 773-780, nov. 2016, doi: 10.1016/j.foodres.2016.10.007.

[5] Engeseth, N. J., & Pangan, M. F. A. (2018). Current context on chocolate flavor development—A review. Current opinion in food science, 21, 84-91.

[6] De Vuyst, L., & Leroy, F. (2020). Functional role of yeasts, lactic acid bacteria and acetic acid bacteria in cocoa fermentation processes. FEMS Microbiology Reviews, 44(4), 432-453.

[7] S. Sood et H. Singh, « Computer Vision and Machine Learning based approaches for Food Security: A Review », Multimed. Tools Appl., vol. 80, no 18, p. 27973-27999, juill. 2021, doi: 10.1007/s11042-021-11036-2.

[8] Ghorbanzadeh, O., Blaschke, T., Gholamnia, K., Meena, S. R., Tiede, D., & Aryal, J. (2019). Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. Remote Sensing, 11(2), 196.

[9] M. M. Oliveira, B. V. Cerqueira, S. Barbon, et D. F. Barbin, « Classification of fermented cocoa beans (cut test) using computer vision », J. Food Compos. Anal., vol. 97, p. 103771, avr. 2021, doi: 10.1016/j.jfca.2020.103771.

[10] B. Khagi, C. G. Lee, et G.-R. Kwon, « Alzheimer's disease Classification from Brain MRI based on transfer learning from CNN », in 2018 11th Biomedical Engineering International Conference (BMEiCON), Chiang Mai, nov. 2018, p. 1-4. doi: 10.1109/BMEiCON.2018.8609974.

[11] A. P. A. C. Barbon, S. Barbon, R. G. Mantovani, E. M. Fuzyi, L. M. Peres, et A. M. Bridi, « Storage time prediction of pork by Computational Intelligence », Comput. Electron. Agric., vol. 127, p. 368-375, sept. 2016, doi: 10.1016/j.compag.2016.06.028.

[12] Muthulakshmi. A et P. N. Renjith, « Classification of Durian Fruits based on Ripening with Machine Learning Techniques », in 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, déc. 2020, p. 542-547. doi: 10.1109/ICISS49785.2020.9316006.

[13] B. Harel, Y. Parmet, et Y. Edan, « Maturity classification of sweet peppers using image datasets acquired in different times », Comput. Ind., vol. 121, p. 103274, oct. 2020, doi: 10.1016/j.compind.2020.103274.

[14] M. Elleuch, R. Maalej, et M. Kherallah, « A New Design Based-SVM of the CNN Classifier Architecture with Dropout for Offline Arabic Handwritten Recognition », Procedia Comput. Sci., vol. 80, p. 1712-1723, 2016, doi: 10.1016/j.procs.2016.05.512.

[15] O. Alharbi, « A Deep Learning Approach Combining CNN and Bi-LSTM with SVM Classifier for Arabic Sentiment Analysis », Int. J. Adv. Comput. Sci. Appl., vol. 12, no 6, 2021, doi: 10.14569/IJACSA.2021.0120618.

[16] M. Usman, S. Ahmed, J. Ferzund, A. Mehmood, et A. Rehman, « Using PCA and Factor Analysis for Dimensionality Reduction of Bio-informatics Data », Int. J. Adv. Comput. Sci. Appl., vol. 8, no 5, 2017, doi: 10.14569/IJACSA.2017.080551.

[17] Kowsari, Jafari Meimandi, Heidarysafa, Mendu, Barnes, et Brown, « Text Classification Algorithms: A Survey », Information, vol. 10, no 4, p. 150, avr. 2019, doi: 10.3390/info10040150.

[18] X. Qi, T. Wang, et J. Liu, « Comparison of Support Vector Machine and Softmax Classifiers in Computer Vision », in 2017 Second International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Harbin, déc. 2017, p. 151-155. doi: 10.1109/ICMCCE.2017.49.

[19] A. Chadha, A. Abdullah, et L. Angeline, « A Comparative Performance of Optimizers and Tuning of Neural Networks for Spoof Detection Framework », Int. J. Adv. Comput. Sci. Appl., vol. 13, no 4, 2022, doi: 10.14569/IJACSA.2022.01304102.

[20] M. Ahmad, S. Aftab, M. Salman, N. Hameed, I. Ali, et Z. Nawaz, « SVM Optimization for Sentiment Analysis », Int. J. Adv. Comput. Sci. Appl., vol. 9, no 4, 2018, doi: 10.14569/IJACSA.2018.090455.

[21] Müller, K. R., Mika, S., Tsuda, K., & Schölkopf, K. (2018). An introduction to kernel-based learning algorithms. In Handbook of Neural Network Signal Processing (pp. 4-1). CRC Press.

[22] S. Ghosh, A. Dasgupta and A. Swetapadma, "A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification," 2019 International Conference on Intelligent Sustainable Systems (ICISS), 2019, pp. 24-28, doi: 10.1109/ISS1.2019.8908018.