International Journal of Modern Achievement in Science, Engineering and Technology (IJSET)
ISSN 3023-459X
DOI: 10.63053/ijset.83

A novel approach to facial recognition utilizing hidden Markov models in computer vision employing Python

Saeed Siravand[1], Dr. Mohammad Hossein Shafie Abadi[2]

1.2. Master's Degree, Islamic Azad University Central Branch, Specialization in Robot Design and Mechatronic Systems,Department of Computer Engineering, Islamshahr Branch, Islamic Azad University, Islamshahr, Iran

ARTICLE INFO

Keywords:
Machine Vision, Face Recognition, Hidden Markov Model (HMM)

ABSTRACT

Image processing today is more commonly referred to as digital image processing, which is a branch of computer science that deals with the processing of digital signals representing images captured by digital cameras or scanned by scanners. In its specific sense, image processing refers to any form of signal processing where the input is an image. The face plays a fundamental role in the identification of individuals and the expression of their emotions within society. Human capability in face recognition is remarkable; we can identify thousands of faces learned throughout our lifetimes at a glance. The CMPA framework is applied to experiments that involved part of a face recognition competition. Analyses indicate that for matching frontal faces in static images, algorithms consistently outperform humans. In the case of video and pairs of static faces, humans are superior. Ultimately, based on the CMPA framework, we proposed a performance index for face recognition, presenting a competitive issue for algorithms that exceed human capabilities in general face recognition tasks. The HMM method relies on matching image templates to a chain of states in a doubly hidden stochastic model. This section addresses the core principles of HMM and describes how to utilize it for face recognition by evaluating training data extraction and the resulting features. It is observed that each segment presents a feature (nose, eye, forehead, etc.). The use of hidden Markov model significantly improves identification rates.

Introduction

In this article, facial recognition plays a fundamental role in identifying individuals and displaying their emotions at the societal level. The human ability to recognize faces is remarkable, as we can identify thousands of previously learned faces throughout our lives at a glance. For example, the capability to model a specific face and distinguish it from a vast number of stored facial models is noteworthy. Face recognition aims to increase the accuracy of identification by employing specialized face vectors, utilizing neural networks and Support Vector Machines (SVM) based on five facial features. Initially, SVM is used to divide the data into two parts, which significantly aids in reducing the workload of the subsequent step, namely face identification using neural networks. This is due to the use of methods with low computational overhead and high accuracy. The proposed method maps the space formed by special faces to the covariance matrix of initial images. After creating the facial space and extracting features, artificial neural networks are utilized for classification. The utilization of neural networks has substantially improved recognition rates. Face retrieval is an important research topic in image processing, aimed at extracting facial images similar to a given photograph (LBP). In this article, we propose a method for retrieving facial images through the combination of histogram of oriented gradients and local binary patterns. The combination of these two methods enhances resistance against variations in facial images, thereby improving the system's performance in image retrieval. To enhance the system's capabilities, we introduce a feedback-driven framework based on SVM in two modes: with and without occluded images. Experimental results on the SVM database indicate that our proposed method can effectively retrieve facial images. Furthermore, comparisons with some successful facial description methods under the MAP framework show an average accuracy of 68%, while the best rate among compared methods reaches 94%. The first and second experimental rates are 61% and 40% respectively. These results demonstrate that our proposed method outperforms existing methods and serves as an effective approach for 90% and 37.99% facial image retrieval. Since 2005, the efficiency of humans and computers has been regularly assessed in the realm of face recognition. This article reviews key results from face recognition algorithms and the human-machine competition. For efficiency analysis during the experiments, the CMPA framework has been utilized. The CMPA framework is applied to experiments that were part of a recognition competition. Analyses indicate that for matching frontal faces in still images, algorithms outperform humans without exception. Conversely, in the case of video and challenging still image pairs, humans demonstrate superiority. Ultimately, based on the CMPA framework, we have formulated a facial efficiency index as a competitive benchmark for algorithms that surpass human performance in general face recognition tasks. The features of new processing units dedicated to visual graphic tasks render them significantly more efficient than traditional Central Processing Units (CPUs) in the execution of complex algorithms. This article examines existing algorithms related to human identity recognition based on three-dimensional facial images. All these algorithms have been compared regarding their accuracy, efficiency, and sensitivity to noise. In conclusion, suggestions have been made concerning the combination of existing algorithms to present more precise and efficient algorithms, providing researchers in the field of human identity recognition from three-dimensional facial images with a clearer perspective. The identification of facial orientation is a crucial research issue in human-computer interactions. To enhance identification accuracy, a new hybrid approach known as the Wavelet Neural Network - Particle Swarm Optimization (SPSO-WNN) has been proposed. In this model, we have recently utilized a proposed particle swarm optimization algorithm (SPSO) to optimize the parameters of the wavelet neural network (WNN). The SPSO-WNN method offers a high convergence speed and superior learning capability compared to WNN, particularly through the introduction of a state equation dependent on the speed updates and Markov changing parameters inherent to SPSO.

1- Face Recognition

Face recognition is a pattern recognition process specifically performed on faces. This process involves categorizing a face as "known" or "unknown" after comparing it with the stored faces of recognized individuals. Computational models of face recognition must address several challenging issues. This difficulty arises from the fact that faces must be represented in a way that best utilizes the information contained in the face to distinguish a specific face from others. In this regard, faces pose a difficult problem as all faces share similar features such as eyes, nose, and mouth.

2-2-1- General Overview of a Typical Face Recognition System

Figure 1 shows the general overview of a typical face recognition system. This overview encompasses features of the general pattern recognition system discussed previously.

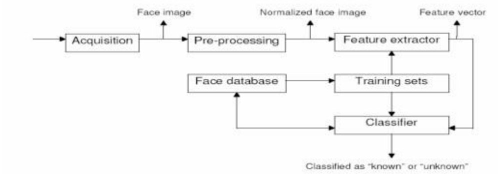

Figure 1 - A General Face Recognition System

There are six main operational blocks, the responsibilities of which are outlined below:

-Reception Module: This is the entry point for the face recognition process. It is the module where the desired facial image is presented to the system. In other words, in this module, the user is requested to provide a facial image to the face recognition system. A reception module can accept facial images from several different environments: a facial image may be a picture file located on a disk or it may be sourced from a magnetic frame grabber, or it could be scanned from a paper document using a scanner.

-Pre-processing Module: In this module, images of faces are normalized through visual techniques and, if necessary, are enhanced to improve recognition system efficacy. Some or all of the following pre-processing steps may be implemented within a face recognition system.

-Image Size Normalization: This process is typically performed to adjust the size of the captured image to a standard size, such as 128, which is the size of the image with which the face recognition system operates.

-Histogram Equalization: This is often applied to very dark or very bright images to improve image quality and enhance face recognition efficacy by adjusting the contrast, thereby making certain significant facial features more apparent.

-Median Filtering: For images with noise, especially those captured via a photographic frame grabber, median filtering can clean the image without losing relevant information.

-High-pass Filtering: Feature extractors based on the overall face may yield better results on images that have undergone edge detection. High-pass filtering emphasizes details such as edges, consequently enhancing edge recognition efficacy.

-Background Removal: To obtain pure facial information, the background surrounding the face can be eliminated. This is particularly important for face recognition systems that utilize information present in the entire image. It should also be noted that in background removal, the pre-processing module must be capable of defining the facial boundary.

-Rotational and Translational Normalization: In certain cases, processing may be required on a facial image where the subject's head is rotated or translated. The position of the head plays a crucial role in determining facial characteristics. Especially for face recognition systems that rely on frontal images, it may be desirable for the pre-processing module to identify rotations or translations and, if possible, to normalize such movements.

-Lighting Normalization: Facial images captured under varying lighting conditions can hinder face recognition efficacy, particularly for face recognition systems based on principal component analysis, where the entirety of the image data is utilized for recognition.

-Feature Extraction Module: Following pre-processing (if deemed necessary), the normalized facial image is forwarded to the feature extraction module to identify key features intended for classification. In other words, this module is responsible for generating a feature vector that accurately represents the facial image.

-Classification Module: In this module, an algorithmic classifier compares the extracted features from the facial image with those stored in the facial database or face library. After completing this comparison, the facial image is classified as either "unknown" or "recognized".

-Training Set: Training sets are utilized during face recognition. The feature extraction and classification modules adjust their parameters using training sets to achieve optimal recognition efficacy.

-Facial Database: Once a face is categorized as "unknown," facial images can be added to the database along with their corresponding feature vectors for future comparisons. The classification module directly utilizes the facial database.

The feature extraction and classification modules are the two components that distinguish the majority of different face recognition systems. It can be posited that the effectiveness of face recognition systems in identifying faces is significantly influenced by the performance of these two modules. In subsequent sections, we will discuss four distinct methodologies employed by face recognition systems within these two modules.

3- Hidden Markov Model

The HMM method is based on matching image templates to a sequence of states within a doubly hidden stochastic model. This section discusses the fundamental principles of HMMs and also elucidates how to utilize them for facial recognition. This section is organized as follows: it begins with an overview of HMMs. Subsequently, some applications of HMMs in machine vision are briefly examined, and finally, an HMM-based architecture for facial recognition is described .

Introduction to HMMs

Generally, HMMs are employed for modeling stochastic vectors of non-stationary time series. They have a clear and immediate application in signal processing, particularly in recognition tasks, wherein the signal of interest is naturally presented as a sequence of spectral estimates that vary over time. Rabiner provides a comprehensible tutorial on HMMs, which will be discussed in subsequent sections .

Definition of a One-Dimensional HMM

An HMM serves as a statistical model for a set of sequences of observations. In speech applications, the observations are sometimes referred to as frames. A specific sequence of observations of length T is represented as oT……01. An HMM consists of a sequence of states numbered from 1 to N, which can be better understood as a generator of observations. States are interconnected by arcs, and each time a state J is entered, an observation is produced according to the Gaussian distribution bj(ot), multivariate with mean μ and covariance matrix Vj associated with that state. The arcs themselves possess their corresponding transition probabilities. The transition from state i to state j has a probability of aij. The probability of beginning the model in state μ, j is defined. Therefore, an HMM

is characterized by the following set of parameters :

N is the number of states in the model.

$$A = \{a_{ij} : 1 \leq i, j \leq N\}$$ which is the transition state matrix.

$$B = \{b_j(\cdot) : 1 \leq j \leq N\}$$ which is the probability function of exiting from the state.

$$\Pi = \{\pi_j : 1 \leq j \leq N\}$$ The probability distribution of the initial state.

In the abbreviated notation, the given model is summarized as λ = { N, A, B, Π }.

Training the model and recognition

For a given model λ, the probability of the sequential occurrence of a state sequence Q = q1 … qT and the corresponding observation sequence O = o1 … oT is calculated by multiplying each transition probability by each emission probability at each stage t as follows.

$$P(\mathbf{O}, Q|\lambda) = \pi_{q_1} b_{q_1}(\mathbf{o}_1) \left[\prod_{t=2}^{T} a_{q_{t-1},q_t} b_{q_t}(\mathbf{o}_t)\right]$$

In practice, the sequence of states remains unknown, and the above equation cannot be computed. Nevertheless, P(O | λ) can be calculated by summing all possible state sequences.

$$P(\mathbf{O}|\lambda) = \sum_{Q} P(\mathbf{O}, Q|\lambda)$$

The key attraction of Hidden Markov Models (HMMs) lies in the existence of a straightforward procedure for maximizing the parameters λ. This procedure is commonly referred to as the Baum-Welch re-estimation and is credited to Baum. It relies on the forward-backward algorithm, wherein the forward probability P(ot+1 … oT | qt = j, λ) and the backward probability P(o1 … ot, qt = j | λ) are efficiently defined through a simple iteration. The forward and backward variables (αt(j) and βt(j)) are defined as follows:

$$\begin{aligned}\alpha_t(j) &= P(\mathbf{o}_1 \ldots \mathbf{o}_t, q_t = j|\lambda)\\ \beta_t(j) &= P(\mathbf{o}_{t+1} \ldots \mathbf{o}_T|q_t = j, \lambda)\end{aligned}$$

Subsequently, the variables can be derived through assignment from the general relationship.

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i) = \sum_{i=1}^{N} \beta_1(i)$$

Python code

```
import numpy as np
from hmmlearn import hmm
from sklearn.decomposition import PCA
```

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score

Step 1: Load image data (for example, using standard datasets such as LFW)from
sklearn.datasets import fetch_lfw_people

# Loading Data
lfw_people = fetch_lfw_people(min_faces_per_person=70, resize=0.4)
X = lfw_people.data # Features (Images in Vector Form)
y = lfw_people.target # Labels (Identities of Individuals)

Step 2: Dimensionality Reduction Using PCA
n_components =   100# Number of features after dimensionality reduction
pca = PCA(n_components=n_components, whiten=True).fit(X)
X_pca = pca.transform(X)

Step 3: Converting Labels to Integers
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

Data Division into Training and Testing Sets
X_train, X_test, y_train, y_test = train_test_split(X_pca, y_encoded, test_size=0.2,
random_state=42)

Phase 4: Construction of the Hidden Markov Model
n_classes = len(np.unique(y_train))  Number of Classes (Individuals)
hmm_models[] =

for person in np.unique(y_train):
Filtering data pertinent to each individual.
 X_train_person = X_train[y_train == person]

# Construction and Training of the Model HMM
   model = hmm.GaussianHMM(n_components=4, covariance_type='diag', n_iter=100)
   model.fit(X_train_person)
   hmm_models.append((model, person))

Step 5: Face Detection Utilizing the Trained Model
def predict(X):
   scores[] =
   for model, person in hmm_models:
      score = model.score(X)
      scores.append((score, person))
   return max(scores)[1]
```

```
# Prediction on Experimental Data
y_pred[] =
for x in X_test:
   pred = predict(x.reshape(1, -1))  # reshape Data for the model.HMM
   y_pred.append(pred)

# Calculation of Model Accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy * 100:.2f}%")
```

1. Data: We utilized the standard LFW (Labeled Faces in the Wild) dataset, which comprises images of various faces .
2.PCA: Principal Component Analysis (PCA) was employed to reduce the dimensionality of the image data and expedite processing .
3.HMM: A Hidden Markov Model (HMM) was constructed for each individual. Subsequently, for every new image, the identity of the person is determined through the comparison of likelihood scores .
Improvements: This implementation serves merely as a simplistic example of the HMM model, and for more complex systems, advanced methods such as neural networks or Support Vector Machines (SVM) should be utilized.

HMMs in Vision

HMMs have been extensively utilized in speech recognition applications, which have been thoroughly studied in this context and are considered a well-established technique in this field. As previously mentioned, HMMs model the statistical characteristics of one-dimensional observed sequences, and speech data is inherently one-dimensional along the time axis.

He and Kundu combined continuous density HMMs with data models for classifying closed two-dimensional shapes. Each shape was represented using an autoregressive sequence derived from the radial distances of the shape's centroid, as illustrated in Figure 1. Each radial distance was predicted using a linear combination of M previous radial distances, in addition to a constant term and an error term. The sequence was divided into T segments of L elements. Each radial feature vector included m autoregressive coefficients for the current radial distance, a constant-to-error ratio, and the mean of the current segment. They conducted experiments with eight classes of objects and trained a specific HMM for each class using 20 training samples. For testing, additional samples from more than 10 classes were utilized, with object recognition accuracy reported as high as 100%.

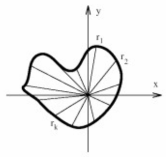

Figure 1 - Radial Sampling Technique

The application of discrete Hidden Markov Models (HMMs) was examined by Kundu for handwritten recognition tasks. An HMM was constructed to model the alphabet.

4-Description of an Architecture

The issue of face recognition is analyzed from the perspective of statistical pattern recognition. Typically, a face can be divided into several regions such as the mouth, eyes, nose, and so forth. If the locations of these regions could be reliably determined, standard pattern matching techniques could be employed for each region separately in order to calculate a comprehensive distance metric. However, in practice, the localization of regions is quite challenging. A possible solution to the aforementioned problem is to establish a connection between the facial regions and the states of continuous density HMMs. By undertaking this approach, it becomes feasible to represent the boundaries between regions through potential transitions between states, and a real image within a region can be modeled using a multivariate Gaussian distribution. Overall, HMMs necessitate a two-dimensional framework. Nevertheless, subsequently, we will resort to a first-order approximation in conditions where the facial regions are restricted to horizontal bands.

Ergodic HMMs

In Ergodic models, each state is accessible from any other state. This implies that all coefficients of the transition matrix A are positive. Generally, Ergodic models are utilized when constraints are applicable to signals and represent the most general type of HMM. To elucidate their operation, an Ergodic HMM is constructed for a sampled image using Figure 2, which scans a P*L window of the image from left to right and from top to bottom. As the sampling window moves left to right along a line, each observation overlaps with Q columns of the previous observation. Once the window reaches the end of the current line, it returns to the start of that line and shifts downward until it overlaps with the upper window of M lines from the previous line.

Each observation oi comprises the intensity levels of the pixels sampled through the window, which are arranged in a column vector.

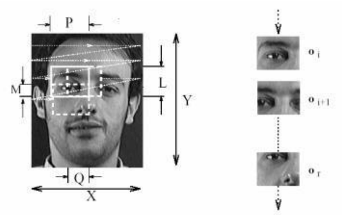

Figure 2 - Sampling Technique for Ergodic Hidden Markov Models

Figure 3 illustrates the data utilized for training an ergodic model and the average distributions of the states of the Hidden Markov Model (HMM) post-training. In this instance, the model comprises 8 states, with the assumption that the forehead, eyes, nose, mouth, chin, cheeks, and two adjacent regions of the face each represent a distinct state. The representation of the average state distributions may assist in understanding how the image is segmented and which features are learned. However, the image on the right side of Figure 2 does not exhibit any recognizable features. Part of the reason for this is the use of the ergodic model without any constraints on the data, resulting in no utilization of structural information. The subsequent section presents a hierarchical model and demonstrates how structural information is utilized, with the state distributions providing features that are discernible to humans.

Figure 3 - Training Data and Model Averages for Ergodic Hidden Markov Model

HMMs with upward-downward states

The states of a Hidden Markov Model (HMM) in an ergodic model can be interconnected arbitrarily, as discussed in the previous section. However, for pattern recognition applications, it is preferable to impose constraints on the permissible state transitions to reflect the characteristics of the data. In particular, left-right topologies of HMMs often exhibit the property that the state index must increase monotonically over the course of processing through the observation sequence. For facial recognition, the natural order involves scanning the face from top to bottom; thus, the upward-downward design is a more natural representation than the left-right orientation. Figure 4 illustrates a sampling technique employing a top-to-bottom window scanning of the face. The observation sequence O from the image X*Y utilizes a sampling window of X*L pixels, which, as depicted in the figure, overlaps with X*M pixels.

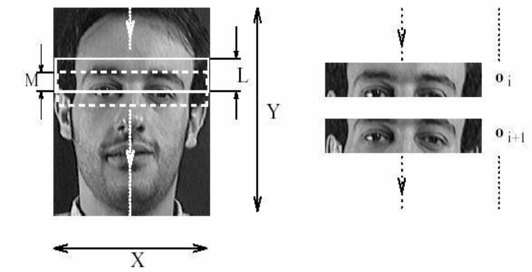

Figure 4 depicts the sampling technique for a hidden Markov model (HMM) with an upward-downward structure.

In Figure 5, you can observe a HMM with an upward-downward configuration that encompasses five states.

Figure 5 - HMM up-down with five states

The extraction of features and segmentation of the training data in this model is illustrated in Figure 6. Observe the right side of the figure and review the obtained features. It is evident that each segment presents a specific feature (such as nose, eye, forehead, etc.).

Figure 6 - Segmented training data and mode averages for the HMM Up-Down

5-Conclusion and Recommendations

The face plays a fundamental role in the identification of individuals and the expression of their emotions within society. The human ability to recognize faces is remarkable; we can identify thousands of faces learned throughout our lives at a single glance. Since 2005, the performance of both humans and computers has been continuously evaluated as part of facial recognition competitions. This article reviews the key results of facial recognition algorithms and the competition between humans and machines. To analyze performance during the experiments, we utilized the CMPA framework. The CMPA framework is applied to experiments that were part of a face recognition competition. The analyses indicate that for matching frontal faces in still images, the algorithms perform consistently better than humans. However, for video and pairs of still facial images, humans excel. Ultimately, based on the CMPA framework, we developed a facial performance index, a competitive issue for algorithmic advancements that surpass human capability in general facial recognition tasks. Facial recognition is a pattern recognition task specifically performed on faces. This task involves categorizing a face as "recognized" or "unrecognized" after comparison with stored faces of known individuals. The HMM method for template matching is based on a series of states in a hidden Markov model. This section examines the fundamental principles of HMM and elaborates on how it can be employed for facial recognition. Feature extraction and the segmentation of training data are evaluated within this model, and the resulting features are observed. It is evident that each segment presents a specific feature (e.g., nose, eye, forehead, etc.).

6- References

1. Amin Islami, Tala, Gol Mohammadzadeh, Neda, Face Recognition Using Image Processing, Second International Conference on New Research Findings in Electrical Engineering and Computer Science, 2016 .

2.Ghassari, Mohammad, Ebrahimnejad, Hossein, Face Image Retrieval Using Histogram of Oriented Gradients and Local Binary Patterns, Journal of Machine Vision and Image Processing, Issue One, 2013 .

3.Rabani, Alireza, Roustayi, Rasoul, Zohrevandi, Abadollah, A Case Study on the Comparison of Face Detection Algorithms Between Humans and Computers, Second National Conference on Computer Engineering Research, 2016 .

4.Sahraei, Reza, Reza Zadeh Moghadam, Ahmad, Modares Shadmehri, Mohammad, Human Identity Detection and Image Processing Algorithms, Second International Conference on Management, Information Technology, and Communications, 2016 .

5. Samadiar, Zahra, Paykhar, Hamid, Combination of Wavelet Neural Network and Particle Swarm Optimization Algorithm for Face Orientation Detection, Second National Conference on Novel Approaches in Computer and Electrical Engineering, 2016 .

6. C.J. Kuo, T.-G. Lin, R.-S. Huang, and S.F. Odeh, "Facial Model Estimation from Stereo/Mono Image Sequence," IEEE Transactions on Multimedia, Vol. 5, No. 1, pp. 8-23, March 2003 .

7.J. Fan, D.K.Y. Yan, A.K. Elmagarmid, and W.G. Aref, "Automatic Image Segmentation by Integrating Color-Edge Extraction and Seeded Region Growing," IEEE Transactions on Image Processing, Vol. 10, No. 10, pp. 1454-1466, October 2001 .

8.L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of the IEEE, 77(2):257-286, 1989 .

9.L.E. Baum, "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions ".