Image and Vision Computing 29 (2011) 693–705

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A mutual information based face clustering algorithm for movie content analysis ☆

N. Vretos *, V. Solachidis [1], I. Pitas [1]

Department of Informatics, University of Thessaloniki, Thessaloniki 54124, Greece

ARTICLE INFO

Article history:
Received 13 December 2010
Received in revised form 15 July 2011
Accepted 29 July 2011

Keywords:
Face clustering
Mutual information
Normalized cuts
Spectral graph analysis
Image processing

ABSTRACT

This paper investigates facial image clustering, primarily for movie video content analysis with respect to actor appearance. Our aim is to use novel formulation of the mutual information as a facial image similarity criterion and, by using spectral graph analysis, to cluster a similarity matrix containing the mutual information of facial images. To this end, we use the HSV color space of a facial image (more precisely, only the hue and saturation channels) in order to calculate the mutual information similarity matrix of a set of facial images. We make full use of the similarity matrix symmetries, so as to lower the computational complexity of the new mutual information calculation. We assign each row of this matrix as feature vector describing a facial image for producing a global similarity criterion for face clustering. In order to test our proposed method, we conducted two sets of experiments that have produced clustering accuracy of more than 80%. We also compared our algorithm with other clustering approaches, such as the k-means and fuzzy c-means (FCM) algorithms. Finally, in order to provide a baseline comparison for our approach, we compared the proposed global similarity measure with another one recently reported in the literature.

1. Introduction

Face clustering is a very important task for movie semantic extraction. It can contribute in many ways, like determining the principal actors or the creation of database references or dialog detection and many others. Moreover, face clustering can be used for unsupervised training of face recognition algorithms and in general as a preprocessing step in any human based image and video processing tasks, so as to create a human wise categorization of the data.

Facial image clustering, put together facial images that belong to the same person by employing a certain image similarity criterion. Let P be a set of facial images. A clustering $\mathcal{C}=\{C_i|C_i\subseteq P\}$ is a division of P into facial image clusters C_i, for which the following conditions hold: $\cup_{C_i\in\mathcal{C}} C_i = P$ and $\forall C_i, C_j \in \mathcal{C}: C_i \cap C_{j\neq i} = \varnothing$. Ideally, the clustered facial images should belong to the same person. Face clustering is a very important application and can contribute in many ways to semantic movie analysis, e.g., for determining the movie cast or for assisting automatic dialog detection. Until now, few face clustering algorithms have been reported in the literature [1–4].

Face recognition and face clustering are two different tasks: in face recognition, we assume that we have a known number of persons and a training facial image database, consisting of certain labeled facial images per person. This database is used for training a face recognition classifier. Then, if we have a test video, each facial image extracted from a video frame can be tested by the already trained face recognition classifier and the best matching person id (or rather a list of best matching people ids) is returned. In face clustering, the number of persons appearing in a video clip or movie is unknown and there is no training facial image database. Therefore, no training is possible. The face clustering goal is entirely different from that of face recognition: given a number of video frames containing facial images, we have to find the unknown number of persons appearing therein, based on facial image similarities. Both face recognition and face clustering may share certain tools (e.g. image similarity measures, face representation methods), but are different in many aspects in terms of goals, methodology (training/no training) and performance metrics. Although, a great amount of work has been conducted on face recognition, face clustering is a rather novel topic with few publications in the literature so far [1–4]. In [2] the authors have proposed an approach for face clustering in video that involves the so called Joint Manifold Distance (JMD). Therein, the authors propose a method, where each subspace represents a set of facial images of the same person detected in consecutive frames. The clustering algorithm, uses a facial video sequence to sequence distance and follows an agglomerative strategy. Another distance metric for clustering and classification algorithms, called Affine Invariant Distance Measure (AIDM) was proposed in [3]. This distance function, which is invariant to affine transformations, is used in combination with partitioning-based algorithms for face clustering. In [4], Foucher et al. recommended a face clustering method based on face detection and tracking

☆ This paper has been recommended for acceptance by Ioannis A. Kakadiaris.

* Corresponding author. Tel./fax: +30 2310996304.
E-mail addresses: vretos@aiia.csd.auth.gr (N. Vretos), pitas@aiia.csd.auth.gr (I. Pitas).
[1] Tel./fax: +30 2310996304.

0262-8856/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2011.07.006

and use several spectral graph techniques for classification. Finally, in our previous work [1], we have proposed a mutual information (MI) based technique for face clustering by constructing a similarity matrix based on the image intensities and have clustered this similarity matrix by means of a fuzzy c-means classifier. The motivation to employ MI comes from its numerous uses as an image similarity measure in other image analysis tasks, e.g. in medical image registration [5], shot cut detection [6], and object tracking [7].

In many applications involving facial images, color spaces are exploited in order to better characterize the facial features. In [8], a specific color space is used for face recognition. In the proposed method, the HSV color space of a facial image (more precisely, only the hue and saturation channels) is used, in order to calculate the mutual information similarity matrix of a set of facial images. We make full use of the similarity matrix symmetries, so as to lower the computational complexity of the mutual information calculation. Thereafter, we assign each row of this matrix as feature vector describing a facial image for producing a global similarity criterion for face clustering. Finally, spectral graph clustering of the global similarity matrix is used to perform clustering.

Spectral graph clustering has been used in image segmentation [9], object recognition [10] and graph-matching [11]. In [12], Carcassoni and Hancock use a coarse-to-fine detail approach, in order to provide a more robust graph clustering process and to overcome problems that arise from spurious graph nodes and edges. In our case, the facial images in a facial image set P can be considered as nodes in a similarity graph, whose edge weights are the facial image similarity. Thus, spectral graph can provide node (i.e. facial image) clustering. As will be demonstrated later on, spectral graph analysis outperforms other clustering methods in face clustering.

The novelty of our approach is primarily in the use of hue and saturation in the calculation of the MI in assessing facial color image similarity, versus the more commonly used image intensity MI [1]. Thus, the proposed method is proven to be robust when, we have facial pose and illumination variations. Moreover, we use a novel feature vector that describes the global similarity of a facial image to the rest of the facial images. This fact provides extra robustness to the proposed method. Finally, spectral graph clustering is applied on the global similarity matrix, which provides superior performance than competing techniques, e.g. k-means or FCM used in [1]. It also outperforms other methods that are used in image registration, mainly due to the fact that such methods are much simpler with respect to light variations and pose and, thus, inappropriate for the face clustering task.

The remainder of this paper is organized as follows: facial color image mutual information and its normalized version (to be used as facial image similarity measures) are presented in Section 2. In Section 3, we present face clustering using N-cuts. In Section 4, we show the face clustering performance metrics and experimental results on two test cases: a) the XM2VTS facial video database [13] and b) another video database coming from extracts of six commercial movies. In the same section, we provide a "baseline" comparison of the employed similarity criterion (i.e. the hue/saturation MI) to another newly developed image similarity criterion [14]. Finally, conclusions are drawn in Section 5.

2. Mutual information for color facial image clustering

Many image similarity measures have been proposed in recent literature [15–21]. An extensive survey of f-measurements and various entropy measures, e.g. the Rèny and Tsalis entropy, are presented in [15]. Other image similarity measures, like the Kullback–Leibler divergence [16] can be used as well. Recent approaches [17–21] to image registration use the mutual information (MI) measure that is proven to be robust under cropping and small illumination perturbations.

The mutual information of two random variables is defined as:

$$I(X,Y) \triangleq \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}, \tag{1}$$

where $p(x,y)$ is their joint pdf and $p(x)$, $p(y)$ are their marginal pdfs. Typically, X, Y represent the image intensity of two different images. The entropy of a random variable X is defined as:

$$H(X) = -\sum_{x\in X} p(x)\,\log p(x). \tag{2}$$

Likewise the joint entropy of two random variables X and Y is defined as:

$$H(X,Y) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\,\log p(x,y). \tag{3}$$

There are several ways of normalizing the mutual information between different pairs of images [22]. The normalized version of mutual information used in this paper is defined as in [17]:

$$N_{MI}(X,Y) \triangleq \frac{H(X)+H(Y)}{2H(X,Y)}, \tag{4}$$

N_{MI} takes values in the domain [0,1]. In [22], Studholme et al. have shown that this version of the normalized mutual information is less sensitive to the size of the overlapping image regions in image registration. A detailed presentation of the aforementioned entropy and mutual information calculation can be found in [23].

In the case of color facial images, we shall use the HSV color space for checking similarity and, in particular, the hue H and saturation S components, which are proven to be robust under illumination changes, in comparison to image intensities [24,25]. In [26], Sobottka and Pitas have shown that face colors occupy a certain region of the HSV color domain. Furthermore, it is proven that, at a specific region of the HS domain, H is the most informative channel [27] for facial colors. Therefore, we employ only the H, S channels of two facial images having hue and saturation values H_1, S_1, H_2, S_2 respectively.

It can be easily shown that the 4D normalized MI is given by:

$$N_{MI}(H_1,S_1,H_2,S_2) = \frac{H(H_1)+H(S_1)+H(H_2)+H(S_2)}{2\cdot H(H_1,S_1,H_2,S_2)}. \tag{5}$$

Let us suppose that the histograms $\hat{p}(h_1)$ and $\hat{p}(h_2)$ to be used in Eqs. (2), (5) have N bins, while $\hat{p}(s_1)$ and $\hat{p}(s_2)$ have M bins. The 4D histogram estimating $\hat{p}(h_1,s_1,h_2,s_2)$ to be used in Eq. (5) has dimensions $N\times M\times N\times M$ and can be found as follows. Let X_1, X_2 be two facial color image regions of interest (ROIs) of size $H\times W$ pixels produced by a face detector/tracker. We transform them in the HSV color space and calculate the 4D joint histogram:

$$\hat{p}\,(h_1,s_1,h_2,s_2) = \frac{1}{H\cdot W}\cdot\left|\{(k,l)\in[1,H]\times[1,W]\,/\,H_1(k,l)=h_1 \;\; and \;\; S_1(k,l)=s_1 \;\; and \;\; H_2(k,l)=h_2 \;\; and \;\; S_2(k,l)=s_2\}\right| \tag{6}$$

where $|\cdot|$ denotes set cardinality and $H_1(k,l)$, $S_1(k,l)$, $H_2(k,l)$, $S_2(k,l)$ are the hue and saturation values for image X_1 and X_2 at pixel (k,l), respectively. Then, $\hat{p}(h_1)$, $\hat{p}(s_1)$, $\hat{p}(h_2)$, $tp(s_2)$, $\hat{p}(h_1,s_1,h_2,s_2)$ and Eqs. (2), (3) are used in calculating Eq. (5). The facial images X_1, X_2 in Eq. (6) must have the same size of $H\times W$ pixels, which is not always true, since face detectors typically produce facial regions of varying size. In order to overcome this problem, we calculate a mean bounding box from the face detector/tracker results on a particular video and scale all facial images to this size. After several experiments, we have concluded that this is the best way to solve the scaling/cropping problems. Other approaches, e.g., scaling each pair of facial image ROIs towards the biggest or the smallest bounding box of the