Pattern Recognition Letters 73 (2016) 1–6

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

A comparative study of data fusion for RGB-D based visual recognition[☆]

Jordi Sanchez-Riera[a], Kai-Lung Hua[b], Yuan-Sheng Hsiao[a,b], Tekoing Lim[a], Shintami C. Hidayati[a,b], Wen-Huang Cheng[a,*]

[a] MCLab, Research Center for Information Technology Innovation (CITI), Academia Sinica, Taipei 115, Taiwan
[b] CSIE, National Taiwan University of Science and Technology, Taipei 106, Taiwan

ARTICLE INFO

Article history:
Received 3 May 2015
Available online 30 December 2015

Keywords:
RGB-D
Fusion
CNN
DBN
SAE
SVM

ABSTRACT

Data fusion from different modalities has been extensively studied for a better understanding of multimedia contents. On one hand, the emergence of new devices and decreasing storage costs cause growing amounts of data being collected. Though bigger data makes it easier to mine information, methods for big data analytics are not well investigated. On the other hand, new machine learning techniques, such as deep learning, have been shown to be one of the key elements in achieving state-of-the-art inference performances in a variety of applications. Therefore, some of the old questions in data fusion are in need to be addressed again for these new changes. These questions are: What is the most effective way to combine data for various modalities? Does the fusion method affect the performance with different classifiers? To answer these questions, in this paper, we present a comparative study for evaluating *early* and *late* fusion schemes with several types of SVM and deep learning classifiers on two challenging RGB-D based visual recognition tasks: hand gesture recognition and generic object recognition. The findings from this study provide useful policy and practical guidance for the development of visual recognition systems.

1. Introduction

Multimodal fusion is an active research topic in multimedia analysis [1,28]. For example, researchers improved the recognition performance by integrating visual features (lips reading), in addition to conventional single modality audio features (voice analysis) for speech recognition or similarly [15] combined different classifiers trained with data from several modalities. Although data fusion has been extensively investigated for audio-visual applications, the availability of new sensory devices capable of capturing synchronized depth and color streams has brought new challenges. In particular, there is still a very much open issue of how, exactly, to fuse depth and color. Moreover, new machine learning techniques, such as *deep learning*, have been shown to be one of the key elements in achieving state-of-the-art inference performances in a variety of applications. However, these new devices and machine learning techniques still raise the same old questions: *What is the most effective way to integrate heterogeneous information from multimodal sensors? Does the design of the fusion method depend on the corresponding applications? Does the employed classification algorithm have an impact on the fusion method and the resultant accuracy?* In this paper, we provide answers to the above questions.

In the literature, *early fusion* and *late fusion* are the two most popular fusion schemes. While *early fusion* approaches integrate data from different modalities before being passed to a classifier, *late fusion* approaches integrate, at the last stage, of the responses obtained after individual features learning the model for each descriptor. Although the employment of fusion schemes is a common technique in audio-visual domains [6,22,25], the works using RGB-D data [13,18,21,30] are still developed through a unimodal fashion, lacking of studies on how to effectively integrate color and depth modalities [2,3,20,30]. In addition, although *deep learning* methods have recently reported promising results when applied to various multimedia applications [11,23,27,31], there is no explicit comparison between the deep architectures and traditional classifiers to explore which is the most suitable classification paradigm for visual recognition with RGB-D data. Typically, in RGB-D applications a depth image is used to segment better the object of interest and then some features are computed for depth and RGB images to afterwards train a classifier [8,9,12]. In contrast, we want to focus on different levels of feature fusion and deep learning classifiers, where an object itself is already localized and segmented from

[☆] This paper has been recommended for acceptance by R. Davies.
[*] Corresponding author. Tel.: +886 2 2787 2356.
E-mail address: whcheng@citi.sinica.edu.tw (W.-H. Cheng).

http://dx.doi.org/10.1016/j.patrec.2015.12.006
0167-8655/© 2015 Elsevier B.V. All rights reserved.

the image, and no pre-processing steps or other machine learning techniques are needed.

Therefore, in this work we conduct a comparative evaluation study of RGB-D visual recognition tasks by assessing the effectiveness of various settings, which include different fusion schemes (e.g., early fusion vs. late fusion) and two state-of-the-art learning mechanisms (e.g., SVM vs. deep learning). To the best of our knowledge, this work is the first to explicitly address the fusion evaluation for RGB-D data with deep learning classifiers.

The rest of the paper is structured as follows. Sections 2, 3, and 4 give details about the adopted fusion methods, classifiers, and recognition tasks, respectively. Section 5 describes how experiments are carried out, and Section 6 draws the conclusions.

2. Fusion schemes

Given RGB-D data from a sensor, a typical operation is to extract features from the data and use the feature based representations to learn a multi-class classifier, i.e., a discriminant function $f:\mathbb{R}^M\times\mathbb{C}\to\mathbb{R}$ where $\mathbb{R}^M$ is the observation space, $\mathbb{C}=\{1,\ldots,C\}$ is the set of labels, and C is the number of classes. A new unlabeled observation $x\in\mathbb{R}^M$ is classified with:

$$c^*(x)=\arg\max_{c\in\mathbb{C}} f(x;c) \tag{1}$$

In this study, the feature vector $\mathbf{x}$ is composed from data of two different modalities, i.e. color and depth. We define $\mathbf{x_{RGB}}$ as the data from a color sensor (RGB) and $\mathbf{x_D}$ as the data from a depth sensor (D). The vector $\mathbf{x_{RGB}}+\mathbf{x_D}$ denotes the concatenation of color features (RGB) and the corresponding depth features (D). Thus, we can have four different options to combine the data for our classifiers:

$$f_{\mathbf{RGB}}=f(\mathbf{x_{RGB}};c) \tag{2}$$

$$f_{\mathbf{D}}=f(\mathbf{x_D};c) \tag{3}$$

$$f_{\mathbf{EARLY}}=f(\mathbf{x_{RGB}}+\mathbf{x_D};c) \tag{4}$$

$$f_{\mathbf{LATE}}=g(f_{\mathbf{RGB}}+f_{\mathbf{D}}) \tag{5}$$

where g is an aggregation function for the output scores of $f_{\mathbf{RGB}}$ and $f_{\mathbf{D}}$. One of the most common aggregation functions is to use a convex weighting scheme as follows:

$$f_{\mathbf{LATE}}=\lambda f_{\mathbf{RGB}}+(1-\lambda)f_{\mathbf{D}} \tag{6}$$

This implies to tune the parameter λ that controls the weight of each of the modalities used. In some algorithms, e.g., Multi-Kernel methods [29], this parameter can be adjusted at the same time of the training phase. In our particular case, to avoid tuning extra parameters we adopt a max function, thus we select the modality with the maximum score.

3. Classifiers

The classifiers adopted in this study are mainly in two different paradigms: *Kernel Method* and *Deep Learning*. The Kernel Method paradigm has been successfully used in the past decades for computer vision tasks such as object recognition and detection. More recently, Deep Learning paradigm based on neural networks has been demonstrated as powerful as the Kernel Method or even better in some cases. Therefore, as a representative of the Kernel Method, the support vector machine (SVM) algorithm is considered. For the Deep Learning, five different models are adopted, including the convolutional neural networks (CNN), *fast region convolutional neural network (F-RCNN)* [7], stacked autoencoders (SAE), deep belief networks (DBN), and restricted Boltzmann machines (RBM).

Traditionally, *Kernel Methods* are associated with features in an attempt to capture most discriminative parts of the image. However, different works in texture classification [14] and gender recognition [19] suggest that raw image is as good or superior as the features approach. Hence, the performance of the *Kernel Method* is evaluated in both raw data and feature vectors. To extract the feature vectors, multiple descriptors have been proposed in the existing literature, e.g., HOG, SURF, SIFT, DAISY, and MSER. We choose the SIFT [4] descriptor because gradient based descriptors have been shown to have certain properties, e.g. scale invariant, among others and this makes the descriptor generally more robust than other types of local descriptors, e.g., color histogram based ones [26].

3.1. Kernel Method

SVM is one of the most commonly used frameworks for classification. This classification method is devised to find the hyper-plane that best separates the given observed data. Hence, the function $f(x;c)$ in SVM has the form:

$$f(x;c)=\sum_{i=1}^{N}\alpha_i K(x,x_i) \tag{7}$$

where $\{x_i\}_{i=1}^N$ is the training set, α_i's are computed during the training phase, and $K(\cdot,\cdot)$ is the kernel function that transforms data onto a higher dimensional space.

We denote $\mathbf{x^{RAW}}$ as the feature vector using raw pixels of either $\mathbf{x_{RGB}}$, $\mathbf{x_D}$, or $\mathbf{x_{RGB}}+\mathbf{x_D}$, and $\mathbf{x^{SIFT}}$ denotes the SIFT descriptor extracted from the corresponding raw data. Thus, the SVM can take a raw image or its SIFT features as the input as follows:

$$SVM^R=f(\mathbf{x^{RAW}};c) \tag{8}$$

$$SVM^S=f(\mathbf{x^{SIFT}};c) \tag{9}$$

3.2. Deep Learning

A neural network is a system inspired by the human brain where a set of interconnected “neurons” (or units) can produce a predicted output from input data. The intermediate layers between the input layer and the output layer are called hidden layers. Therefore, by denoting the number of hidden layers n and the input data x, the output function $f(x;c)$ can be defined as follows:

$$f(x;c)=f_{n+1}(f_n(\ldots f_2(f_1(x))\ldots)) \tag{10}$$

with

$$f_i(x;c)=\sigma_i(W_i\cdot x+b_i), \tag{11}$$

where σ_i is the activation function for the layer i, typically tanh or a sigmoid function for the hidden layer $(i=1,\ldots,n)$ and a Gaussian or *softmax* function for the output layer $(i=n+1)$. Besides, W_i and b_i are the weights and the bias parameter for the layer i, respectively. Thus the five adopted models are:

$$CNN=f(\mathbf{x^{RAW}};c) \tag{12}$$

$$F\text{-}RCNN=f(\mathbf{x^{RAW}};c) \tag{13}$$

$$SAE=f(\mathbf{x^{SIFT}};c) \tag{14}$$

$$DBN=f(\mathbf{x^{SIFT}};c) \tag{15}$$

$$RBM=f(\mathbf{x^{SIFT}};c) \tag{16}$$

Note that, as suggested in [17], CNN and F-RCNN take raw data, and SAE, DBN and RBM take the extracted features as the input.