SmartVisionApp: A framework for computer vision applications on mobile devices

Cristiana Casanova, Annalisa Franco, Alessandra Lumini *, Dario Maio

DISI, Università di Bologna, Via Sacchi 3, 47521 Cesena, Italy

ARTICLE INFO

Keywords:
Computer vision
Content analysis
Image classification
Mobile landmark recognition

ABSTRACT

In this paper a novel framework for the development of computer vision applications that exploit sensors available in mobile devices is presented. The framework is organized as a client–server application that combines mobile devices, network technologies and computer vision algorithms with the aim of performing object recognition starting from photos captured by a phone camera. The client module on the mobile device manages the image acquisition and the query formulation tasks, while the recognition module on the server executes the search on an existing database and sends back relevant information to the client. To show the effectiveness of the proposed solution, the implementation of two possible plug-ins for specific problems is described: landmark recognition and fashion shopping. Experiments on four different landmark datasets and one self-collected dataset of fashion accessories show that the system is efficient and robust in the presence of objects with different characteristics.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Computer vision is the science aimed to understand the content of images or other complex data acquired by means of different kinds of devices and sensors (Morris, 2004); these techniques are extremely useful in a wide range of possible applications, in fields such as medicine, industrial processes and security. Each computer vision application has the goal of gathering useful information based on visual clues extracted from data acquired from static images, video sequences or other data sources. Computer vision is a research area widely studied and investigated in the past decades, but now the growing diffusion of mobile devices with their high processing capabilities and powerful sensors like camera, GPS and compass has renewed the interest in this field, since smartphones and tablets turned to be an ideal platform for computer vision application. Many possible applications could benefit from the capability of computer vision techniques to perform object recognition with mobile devices; some examples are mobile shopping (where the user can be interested in obtaining information about a product), personalized mobile services (e.g. for purchasing tickets), mobile landmark recognition for tourists or people recognition (for tagging, etc.). These themes are central also to the smart city concept and recur in its definitions. A smart city is a high-performance urban context, where citizens become actors, integral part of the system (Schaffers, Komninos, & Pallot, 2012). Among ICT concepts for smart city, the possibility of collecting location- and time-aware data from the urban context and make them available to the citizens is a particularly interesting feature that has been researched in the last years (Calderoni, Maio, & Palmieri, 2012).

In this context a new framework that combines mobile devices, network technologies and computer vision algorithms to address the problem of object recognition from static images is proposed. This framework has been developed at the Smart City Lab (http://smartcity.csr.unibo.it), a research laboratory part of DISI (Department of Computer Science and Engineering) – University of Bologna.

The framework has been designed for general purpose applications involving object recognition on the basis of visual, textual and spatial information. Moreover, to show the effectiveness of the proposed solution, the implementation of two possible plug-ins for specific problems is described: landmark recognition and fashion shopping. The landmark recognition plug-in allows users to discover information about monuments: both exact and approximated queries are possible and detailed information about the monuments is retrieved for each result found. The fashion shopping plug-in performs analogous queries in the context of clothes and accessories. Both plug-ins allow to combine information of different nature (shape, color, textual information, position, …) on the basis of the user preferences. Anyway the searching capabilities of the two plug-ins are quite different, in order to fit the problem requirements. For landmark recognition the search will be directed to find an image representing the same object of the given photo,

* Corresponding author. Tel.: +39 0547339236.
E-mail addresses: cristiana.casanova@studio.unibo.it (C. Casanova), annalisa.franco@unibo.it (A. Franco), alessandra.lumini@unibo.it (A. Lumini), dario.maio@unibo.it (D. Maio).

0957-4174/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.04.037
while for fashion shopping a similarity search is required in order
to discover several objects similar to the given one. For this reason
the vision components implemented for the two plug-ins and de-
scribed in Section 4 use different visual features.

For both applications the system grants two functionalities:

- Image search: The user can discover information about the
  input image by specifying particular searching criteria; this
  functionality involves a comparison between the input image
  and the pictures stored in the database based on proximity, tags
  or visual content;
- System knowledge base update: The user can contribute to pop-
  ulate the knowledge base by adding new contents (images and
  related information).

The rest of this paper is organized as follows. In Section 2, related
work is provided. In Section 3, the architecture of the proposed
system is described in detail; in Section 4 the implementation of
the computer vision components (related to both applications) is
described. Experiments on several benchmark datasets are dis-
cussed in Section 5, and finally conclusions and future research
directions are given in Section 6.

2. Related works

In recent years, with the improvement of image retrieval algo-
rithms, researchers have given more attention to the image search
engine and have introduced several systems for landmark recogni-
tion or object recognitions. In particular many applications have
been developed for mobile devices, where built-in cameras and
network connectivity make them increasingly attractive for users
to take pictures of objects and then obtain relevant information
about them. Most of proposed apps focus on a single problem
and are based on new algorithms for content analysis, landmark
classification or object recognition (Yap, Chen, Li, & Wu, 2010).

The first pioneer studies that used the built-in camera of a mo-
BILE phone for image retrieval proposed the use of existing search-
ing engine to perform the search. In Yeh, Tollmar, and Darrell
(2004) a two-steps system based on images and text is proposed,
where images are first searched on the web, then text extracted
from the webpages is used to query Google text search engine.
Some research reports results obtained on images taken by a phone
camera but processed offline. For example in Fritz, Seifert, Kumar,
and Paletta (2005) a building detection approach is proposed,
based on the i-SIFT descriptors, an “Informative Descriptor Ap-
proach” obtained by extracting color, orientation and spatial infor-
mation for each Scale-Invariant Feature Transform (SIFT) feature.
Other context aware content works that focus on large-scale land-
mark non-mobile recognition are presented in Schindler, Brown,
and Szeliski (2007) and Zamir and Shah (2010).

Some experiments about the use of image retrieval methods to
locate information around the world are reported in Jia, Fan, Xie,
Li, and Ma (2006). Other attempts of using text to improve image
or video retrieval are reported in Sivic and Zisserman (2003) where
the authors use “visual words” for object matching in videos. After
that, other researches focused on finding out the better way to gen-
erate the visual words (Jégou, Douze, Schmid, & Pérez, 2010; Zhang,
Marszalek, Lazebnik, & Schmid, 2007) resulting in a very performing
approach named “bag of visual words” (Sivic & Zisserman, 2009).

As mobile phones with imaging and locating capability are becom-
ing more widespread, several works about mobile land-
mark recognition have been proposed (Bhattacharya & Gavrilova,
2013). In Redondi, Cesana, and Tagliasacchi (2012) an efficient
coding of Speeded Up Robust Features (SURF) descriptors suit-
able for low-complexity devices is proposed, and a comparative
study of lossy coding schemes operating at low bitrate is carried
out. In order to avoid the latency of an image upload over a
slow network, in Hedau, Sinha, Zitnick, and Szeliski (2012) an ur-
ban landmark recognition approach computed on client is pro-
posed: the presented method involves uploading only GPS
coordinates to a server, then downloading a compact location-
specific classifier (trained on few geo-tagged images in the near-
by of the searched object) and performing recognition on the
client. Differently from our method, this approach requires both
visual data and GPS coordinates for the search. Recognition on
client is used also in Takacs et al. (2008) where an outdoors
augmented reality system for mobile phones is proposed. Cam-
era-phone images are matched on board against a database of
location-tagged images using a robust image retrieval algorithm
based on SURF descriptors. Matching is performed against a re-
duced database of highly relevant objects, which is continuously
updated to reflect changes in the environment (according to
proximity to the user).

3. System architecture

The system was designed according to a client–server architec-
ture, where the client is a mobile device and the server is a physical
machine that processes the images sent from the client. A graphical
schema of the software architecture is reported in Fig. 1 for the
functionality of image search and in Fig. 2 for the functionality of
system knowledge base update, respectively.

The client application, designed for Android platforms, performs
the following tasks:

- User mode selection: It allows the user to select the category of
  objects of interest among the existing possibilities; in the pre-
  sent prototype the following choices are available: monuments,
clothes, and accessories.
- Image acquisition: This function allows the user to take a photo
  to or to select an image from the photo gallery.
- Query: The input image is sent to the server. On the basis of the
category selected and of the searching criteria provided by the
user, the server searches the input image in the system knowl-
edge database and sends back the related information. If a
match is not found in the database, the user can exploit the sys-
tem knowledge base update functionality to add the image and
related information.
- Results visualization: The best match or a ranked list of images
  is displayed to the user.

In Fig. 3 a screenshot of the client application for each of the
tasks described above is reported.

The user can select different searching criteria, according to the
category of objects selected. In case of landmark recognition the
following options are available:

- Perform exact or approximated queries without spatial con-
  straints: The landmark in the image (exact) or similar land-
  marks (approximated) are retrieved without any reference to
  spatial coordinates;
- Perform exact or approximated queries with spatial constraints:
The landmark in the image (exact) or similar landmarks
(approximated) are retrieved at a range distance selected by
the user; the GPS coordinates (latitude and longitude) are
exploited to this aim.

For the fashion shopping module the following two options are
available:

- Recognize a given object on the basis of color, shape, or a com-
bination of both.
Search the DB by keywords in order to find objects according to specific searching criteria (stylist, price, shape, etc.)

The server is composed by the following modules:

- The Java server is designed to decode the input image, send it to the computer vision component, retrieve data from the database (querying by specific searching criteria) and send back the result to the client;

- The server is composed by the following modules:
• The computer vision component is devoted to image processing, involving feature extraction and matching against stored data. Since all the main procedures of a computer vision application are strictly problem-dependent, the methods for image preprocessing, feature extraction and classification are not included in the general framework and are considered as a part of the plug-in component.

The communication between client and server is performed according to the following protocols:

• **Socket protocol** is one of the main technologies of computer networking, allowing applications to communicate using standard mechanism built into network hardware and operating systems. This protocol is used for image transferring from mobile device to the server, decoding the input image into a byte array and sending it to the Java server. The use of the socket protocol causes low traffic network and it is faster than the HTTP protocol even if more difficult to manage.

• **HTTP protocol** is used to send parameters and images in order to add a new object to the system database. The update functionality has been implemented using this protocol since it allows a simple communication with the PHP script which performs “system knowledge base update” task.

The internal database is stored in MySQL and designed using phpMyAdmin. A table for each plug-in is needed; the tables for storing data for monument and fashion modules are designed according to schemas in Tables 1 and 2.

If the user performs a search on the basis of specific search criteria, the system performs a simple SQL query combining filters to retrieve the relative information. In this case a similarity search on the input image is not needed. On the contrary, when the user requires a similarity search by using the input image, the searching process is performed by the computer vision components as explained in the following sections. After the ID of the corresponding object is retrieved, a SQL query is executed to obtain the complete information.

The testing application has been developed using the following hardware and software configurations:

• **Client:** The user interface has been designed for Android 2.3 and optimized for Samsung Galaxy S I9000 and Samsung Galaxy SIII I9300.

• **Server:** The server runs on a Window 7 Pro 64bit machine (PC with Intel i5-2500 3.30 GHz 8 GB RAM) with MATLAB R2011a and Java SE 7.

### 4. Computer vision components

A general computer vision system consists of:

#### Table 1

Database schema on the “monuments” table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Numerical</td>
<td>Internal identifier</td>
</tr>
<tr>
<td>Name</td>
<td>Text</td>
<td>Name</td>
</tr>
<tr>
<td>Type</td>
<td>Text</td>
<td>Architectural style</td>
</tr>
<tr>
<td>Architect</td>
<td>Text</td>
<td>Architect</td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td>City where the monument is located</td>
</tr>
<tr>
<td>Latitude</td>
<td>Numerical</td>
<td>GPS latitude</td>
</tr>
<tr>
<td>Longitude</td>
<td>Numerical</td>
<td>GPS longitude</td>
</tr>
<tr>
<td>Description</td>
<td>Text</td>
<td>A short description about the monument</td>
</tr>
<tr>
<td>URL</td>
<td>Text</td>
<td>Link to Wikipedia for detailed information</td>
</tr>
<tr>
<td>Image</td>
<td>Text</td>
<td>Name of the image file</td>
</tr>
</tbody>
</table>

#### Table 2

Database schema on the “clothes/accessories” table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Numerical</td>
<td>Internal identifier</td>
</tr>
<tr>
<td>Stylist</td>
<td>Text</td>
<td>Stylist or designer</td>
</tr>
<tr>
<td>Type</td>
<td>Text</td>
<td>(Jacket, skirt, etc.)</td>
</tr>
<tr>
<td>Color</td>
<td>Text</td>
<td>Color</td>
</tr>
<tr>
<td>URL</td>
<td>Text</td>
<td>Link to Wikipedia for detailed information</td>
</tr>
<tr>
<td>Image</td>
<td>Text</td>
<td>Name of the image file</td>
</tr>
</tbody>
</table>

• Data acquisition and preprocessing, performed in the proposed framework by the client;

• Feature extraction: Characterization of an object by descriptors possibly with high discriminating capability and robustness to possible object variations.

• Matching: Evaluation of the similarity among descriptors in order to recognize the input object or identify it as a member of a predefined class.

In this work the practical implementation of the different components strictly depends on the search functionality (landmark or fashion objects recognition) since the salient features of different object categories usually vary significantly.

#### 4.1. Landmark recognition module

The visual features commonly used for landmark recognition can be divided into two categories (Chen, Wu, Yap, Li, & Tsai, 2009):

• **Global features:** They are based on color, edge and texture. Color is one of the simplest features to recognize landmarks but, unfortunately, it is sensitive to changes in illumination and contrast.

• **Local features:** They are widely used in landmark recognition due to their high capability of describing the region of interest. The most popular approaches for local-features extraction are the keypoint-based technique (Jiang, 2010) and the dense-sampling approach (Nowak, Jurie, & Triggs, 2006). Keypoint-based features, including SIFT (Lowe, 2004), are widely used in this problem because of their robustness to changes in illumination, scale, occlusion and background clutter. The dense-sampling approach extracts visual features from local patches randomly or uniformly driven from the images. One of the most used techniques to represent an image using patch features is the bag-of-word method (BoW) (Csurka, Dance, Lixin, Willamowski, & Bray, 2004). BoW is inspired by models used in natural language processing: the idea is to treat an image as a document and represent it as a sparse histogram over a codebook of “words”: the codebook is obtained by performing vector quantization of the patch descriptors from all the classes, then building a histogram of the codewords as final descriptor.

In this work a local feature approach based on SIFT and BoW is adopted. A graphical schema of our approach is shown in Fig. 4.

The codebook creation process (performed only once during the training stage) can be summarized as follows:

• Collect SIFT features from images: The local patches are located and described using the Scale Invariant Feature Transform (SIFT): it detects keypoints as maxima or minima of a difference-of-Gaussian function; at each point a feature vector is
extracted that is invariant to image translation, scaling, and rotation, and partially invariant to illumination changes and affine or 3D projection.

- Use k-means to cluster the features extracted into a visual vocabulary: Given a set of training images from different landmarks, a vector quantization of features is performed and the k-means algorithm is used to group keypoints from each landmark. In order to limit the number of descriptors in the codebook, each cluster centroid is selected to represent a visual word and is added to the visual dictionary.

The representative visual words selected above are then used for feature extraction (both for representing the images in the database and for describing the query images). Feature extraction consists of building a histogram of the visual word frequencies. Each image is represented as a histogram of codewords by assigning each SIFT descriptor extracted from the image to the nearest word in the dictionary.

The recognition of an input image among a set of stored landmarks is performed by a general purpose classifier trained to distinguish among different landmark templates. In this work Support Vector Machine (SVM) [Cristianini & Shawe-Taylor, 2000] is used, a two-class classifier which constructs a separating hyperplane as the decision boundary using the support vectors from the training sets.

4.2. Fashion shopping module

Due to the excellent properties of invariance to scale, rotation and point of view, the SIFT descriptors, which summarizes information about color, shape and texture, are widely used in object recognition. Nevertheless, the problem of clothes and accessories recognition, requires to perform a discrimination based on colors and contour shape, therefore the SIFT features are not the optimal descriptors to describe the objects. In this work searches based on color, shape and a combination of both have been experimented.

Color moments (Stricker & Orengo, 1995) are popular color descriptors for object recognition due to their simplicity, effectiveness and efficiency. The basis of color moments lays in the assumption that the distribution of colors in an image can be interpreted as a probability distribution: this distribution is characterized by a number of unique moments that can be used as features to identify that image. According to the approach of Stricker and Orengo (1995) three central moments of an image’s color distribution for each color channel (in our case RGB channels) are used: every image is therefore characterized by nine moments (three moments for each three color channels).

Given an image \( I \) of \( N \) pixels, let \( p_{ij} \) be the \( i \)-th color channel at the \( j \)-th image pixel; the three color moments are:

- **Mean**: The average color value in the image

\[
E_i = \frac{1}{N} \sum_{j=1}^{N} p_{ij}
\]  

- **Standard deviation**: The square root of the variance of the distribution:

\[
\sigma_i = \sqrt{\frac{1}{N} \sum_{j=1}^{N} (p_{ij} - E_i)^2}
\]

- **Skewness**: The measure of the degree of asymmetry in the distribution:
$s_i = \left\lfloor \left( \frac{1}{n} \sum_{j=1}^{n} (p_{ij} - E_i)^3 \right) \right\rfloor$

In this application the input image is divided into $3 \times 3$ equal sub-areas and the color descriptors $c \in \mathbb{R}^3$ are extracted separately from each region.

The distance between two images $I_1$ and $I_2$ is the sum of the distance among their descriptors of each region. Several distance functions have been tested to calculate the distance between pairs of regions. As shown in Section 5.3 the best results were obtained using Bhattacharya distance. Bhattacharya distance is widely used in pattern recognition and it is defined as:

$$B(x, y) = \sum_{j=1}^{D} x(j) y(j)$$

where $x, y \in \mathbb{R}^D$ are the feature vectors ($D = 9$ in this case).

The distance values among pairs of images can be used directly to rank results according to their similarity to the searched object, or if the stored objects are classified (i.e. by color or shape), the distance scores can be used to train a general purpose classifier for performing the classification task. Results about both the applications can be found in the experiments.

In the literature several shape descriptors have been proposed for object recognition (Mingqiang, Kidioyo, & Jospeh, 2008), which can be roughly divided into three main categories:

- Contour-based methods and region-based methods, which use shape boundary points.
- Space domain and transform domain, which match shape on point basis (space domain) or on feature basis (feature domain).
- Information preserving (IP) and non-information preserving (NIP), which provide an accurate reconstruction off a shape from its descriptor (IP) or just a partial ambiguous reconstruction (NIP).

In this work the invariant moments (Hu, 1962) have been used, which are considered very popular among the region-based ones. The general form of a moment function $m_{pq}$ of order $(p + q)$ of a shape region can be defined as:

$$M_{pq} = \sum_{x} \sum_{y} \psi_{pq}(x, y)f(x, y)$$

where $\psi_{pq}$ is known as the moment weighting kernel and $f(x, y)$ represents the shape region of an image of $W \times H$ pixels, where

$$f(x, y) = \begin{cases} 1 & \text{inside the region} \\ 0 & \text{outside the region} \end{cases} \quad \text{for} \quad x \in [1 \ldots W], \quad y \in [1 \ldots H]$$

For shape region segmentation the active contour/snake model (Bresson & Esedoglu, 2005) is used: it consists of evolving a contour in images toward the boundary objects. This approach can be seen as a special case of general technique of matching deformable model to an image by means of energy minimization. The approach adopted in this paper is based on Active Contours Without Edges (ACWE) model (Chan & Vese, 2001), which detects boundaries of objects on the basis of homogeneous regions, differently from classical models where large image gradients are employed.

The invariant moments are based on the theory of algebraic invariants as the relative and absolute combinations of moments that are invariant with respect to scale, position and orientation.

The invariant moments can be obtained using central moments, which can be defined as follows:

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \quad p, q = 0, 1, 2, \ldots$$

where $\bar{x} = m_{10} / m_{00}$ and $\bar{y} = m_{01} / m_{00}$.

The point $x, y$ corresponds to the centroid of the image $f(x, y)$. The centroid moment $\mu_{pq}$ is equivalent to $m_{pq}$ if its center has been shifted to the centroid of the image: this is why the central moments are invariant to image translations. To obtain also scale invariance, the normalized moments are used, defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}}, \quad \gamma = \frac{(p + q + 2)}{2}, \quad p + q = 2, 3, \ldots$$

In this work the seven invariant moments proposed by Hu (1962) are used for their properties of invariance to scaling, translation and rotation. The seven invariant moments for shape recognition are based on normalized central moments and are defined as follows:

$$\phi_1 = \eta_{20} + \eta_{02}$$
$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$
$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \mu_{03})^2$$
$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \mu_{03})^2$$
$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[\eta_{30} + \eta_{12}]^2 - 3(\eta_{21} - \eta_{03})^2$$
$$+ (3\eta_{21} - \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$
$$\phi_6 = (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2[4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$
$$\phi_7 = (\eta_{21} - \eta_{03})(\eta_{30} - \eta_{12})(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2$$
$$- (\eta_{30} - 3\eta_{12})(\eta_{21} - \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

According to our experiments the similarity between two invariant moments descriptors has been calculated using Bhattacharya distance as in color moments case.

5. Experimental results

The experimental evaluation of the proposed system has been conducted on four datasets: three well-known benchmarks and a self-collected dataset for landmark recognition and a self-collected dataset of fashion accessories. In our experiments the computer vision components of the system have been extensively tested to evaluate both the efficiency and effectiveness of the recognition system.

5.1. Datasets

For landmark recognition the following four datasets have been used:

- Mobile Phone Imagery Graz (MPG20)\(^1\) (Fritz et al., 2005): It contains 80 images (640 x 480 pixels) of 20 buildings, four views for each building. Few samples from MPG20 are shown in Fig. 5. According to the original protocol, two images of each object taken by a viewpoint change of about ±30° of a similar distance to the object are selected for training and the two additional views (of distinct distance and therefore significant scale change) for testing.

\(^1\) Available at: http://dib.joanneum.at/cape/MPG-20.
• Caltech Building Dataset (Caltech)\(^2\) (Aly, Welinder, Munich, & Perona, 2009): It contains 250 images (1536 x 2048 pixels) of 50 buildings around the Caltech campus. Five different images were taken for each building from different angles and distances. Due to computational issues the images of this dataset have been resized to 480 x 640 pixels. Few samples from Caltech are shown in Fig. 6. According to Hedau et al. (2012), a fivefold cross validation testing protocol has been used on this dataset.

• Zurich Building Dataset (ZuBuD)\(^3\) (Shao, Svoboda, Ferrari, Tuytelaars, & Van Gool, 2003): it contains a training set of 1005 images (480 x 640 pixels) from 201 Zurich city buildings, with five images for each building, with different points of view. Differences among the views include the angle at which the picture is taken, relatively small scaling effects and occlusions. The ZuBuD comes with a standardized query set, consisting of 115 images of buildings occurring in the database: these images are taken with a different camera under different conditions. Few samples from ZuBuD are shown in Fig. 7.

• Italian Landmarks Dataset (ItaLa)\(^4\): It contains 1435 images (of different sizes) from 41 famous Italian buildings. For each building there are 35 images with different points of view, scaling or occlusions. Few samples from ItaLa are shown in Fig. 8. A fivefold cross validation testing protocol have been used on this dataset.

• For the evaluation of the fashion shopping module a self-collected dataset of fashion accessories has been used:

• Fashion Accessories Dataset (FAD)\(^5\): It contains 132 images with different colors and shapes. The images are labeled by colors (11 classes) and shapes (7 classes) into classes with a different number of objects per class. The combined labeling generates a joined classification into 42 non-empty classes. Few samples from FAD are shown in Fig. 9. A twofold cross validation testing protocol have been used on this dataset.

5.2. Experiments on landmark recognition

The Landmark recognition module contains many parameters (i.e. number of words, dimension of the dictionary, parameters of the SVM classifier, etc.) that have to be initialized at the beginning of the procedure itself. These arguments have been optimized on MPG20 dataset, using a 4-fold cross validation.

In Table 3 the recognition accuracy obtained in the four landmark datasets is reported.

The results reported in Table 3 show that the proposed system achieves a very good recognition performance in all the tested datasets and the reported results are comparable with other state-of-the-art approaches. The training time is quite high (13 m for the “ItaLa” dataset on a Intel i5-2500 3.30 GHz) but the search time is very low, so that the retrieval can be performed in real-time.

---

\(^2\) Available at: http://vision.caltech.edu/malaa/datasets/caltech-buildings/.

\(^3\) Available at: http://www.vision.ee.ethz.ch/showroom/zubud/.

\(^4\) Available at: http://bias.csr.unibo.it/lumini/download/dataset/ItaLa.zip.

\(^5\) Available at: http://bias.csr.unibo.it/lumini/download/dataset/fad.zip
Fig. 6. Samples from Caltech dataset.

Fig. 7. Samples from ZuBuD dataset.
Fig. 8. Samples from ItaLa dataset.

Fig. 9. Samples from FAD dataset.
The experiments show that Color and Invariant moments have performance very similar to SIFT descriptors for this problem: the choice of using moments is therefore motivated by computational considerations. The Bhattacharya distance reaches the best performance (also against other distances tested but not reported here for sake of space); the use of a trained classifier as SVM allows a slight improvement, but in our opinion this does not compensate the overhead of a training phase. As concerns the Color&Shape classification problem, the drop of performance is probably due to the fact that several classes have a very low number of samples, therefore the classification is harder than the other two ones. Perhaps an ad hoc tuning of the weighing factors could improve the performance.

6. Conclusions

In this paper a new framework for the development of computer vision applications in mobile devices is proposed; two practical implementations have been tested: landmark recognition and fashion shopping. With quick development of mobile devices, a large number of people already own smartphones. The SmartVisionApp system provides an attracting way to develop image-based searching application using images captured by mobile cameras.

The framework has been designed for general purpose applications involving object recognition on the basis of visual, textual and spatial information. Differently to most of existing approaches our system performs search based on image similarity, without requiring GPS-coordinates. The effectiveness of the framework is tested in a landmark recognition module, a touristic application aimed at discovering information about monuments, and in a fashion shopping application, designed to perform similarity searches in the context of clothes and accessories. Both applications allow combining information of different nature on the basis of the user preferences and grants two functionalities: image search and system knowledge base update. The computer vision modules have been extensively evaluated on several datasets: our experiments demonstrate that both components work well with medium-size databases.

As a future work it is worth to mention the need to make the systems more scalable to deal with larger databases. Several works have faced the problem of reducing the length of SIFT features without losing accuracy. The proposed application requires ad hoc indexing approaches, to be inserted into the visual components, to improve the efficiency of the search. Another possible upgrade of this framework is the possibility of making available as internal components some more general methods for image preprocessing, feature extraction and classification to be used for the design of the plug-in components. Finally the effectiveness and efficiency of the system under more case studies shall be tested and analyzed.

References


Conference on mobile data management: systems, services and middleware (pp. 626–630).


