Book Cover Recognition Project

Carolina Galleguillos
Department of Computer Science
University of California San Diego
La Jolla, CA 92093-0404
`cgallegu@cs.ucsd.edu`

Abstract

The purpose of this project is to recognize book covers from images that are taken by a regular digital camera or a webcam. The recognition will be made through image processing techniques, using SIFT descriptors and transformations, that will allow to identify the correct image that represents the cover from a database of covers. The input images will consist of a book cover and a background were the book was placed. The cover recognition project is base on the Delicious Library[4] for Apple Computers[3], which manages media collections via bar code scanning with an iSight camera, and grabs cover art from Amazon[2].

1   Related Work

There are been several approaches to object recognition. Some project that have similar context are "Video-based Car Surveillance: License Plate, Make, and Model Recognition" by Dlagnekov[8], "Shape Matching and Object Recognition Using Low Distortion Correspondence", A.C. Berg, T.L. Berg and J. Malik[1] and "Learning to detect objects in images via a sparse, part-based representation" by Agarwal[10].

2   Dataset

The dataset to use in the project will be divided in two categories: training set and test set. The first set will be used to train the algorithms and experiment with the keypoints. The second will be used to test the precision and execution of the algorithms. For the final database the dataset will be joint.

2.1   Training Set

Training set will be obtain from Google Print Beta[6], since it gives good resolution of images from book covers (two different sizes) and contains a large amount of images easy to retrieve. Google Print has two different sizes for book covers: a big one, which was scanned from the original book cover (around 575 x 825 pixels) and a small one, with measures around 128 x 183 pixels. Each image have the sentence "Copyrighted Material"

in the right side of the cover which adds a small "noise" to our data. This images compared with image collection of books cover from Amazon have less noise, since the last ones have 2 times the same sentence, at the upper and lower part. Considering resolution of the images, the data set that could be retrieve from the Amazon site have much better quality and images are very sharp. However, Google Print Beta[6] images have the resolution of an average scanner, so quality wise is much lower, but they will be better suited for our input data, since a image from a book will be captured.

When a book cover is not available from Google Print Beta[6], it will be used Amazon[2] book cover for the database, since the sizes are relatively similar, although quality wise may be better sometimes.

The amount of data needed for training the classification algorithm, should be around hundreds (around five hundred), since it would provide different examples of covers to the algorithm, and it would not take too much time to recollect and extract the features from them. The training set should have a good diversity of images, with different colors and designs, that can make the set representative. This training set will also be used as part of the data set to match by the input generated by the user when it captures the book cover (in the database).

2.2 Test Set

The test set will be obtain from the same sources, and it will not include the training set. It will be composed for a different types of book covers that can represent at some extent the diversity of all the book covers that exist. This set will also include book cover that are very similar to each other, in order to test the accuracy of the algorithms.

3 Capturing Images

In order to capture the book covers it will be use a webcam with average resolution. This webcam will allow us to obtain a medium quality picture of the cover. The quality of this picture will present us an more real environment of the capture of the book cover image. We chose to use this low resolution device instead of a good resolution camcorder because the application will be use by users that have access to normal webcams than a camcorder (we assume that most of the users will be prefer the cheap option of a webcam instead an expensive camcorder). The algorithms used in the application will have to deal with lower resolution in order to get a better classification.

The specifications of the webcam to use are:

- Color VGA (640x480) CMOS image sensor.
- High quality lens.
- Focus range of 6 inches to infinity.
- Manual focus.
- Field of view at 44 degrees (Horizontal).
- Attachable to Laptop.
- Captures video in 24bit color.
  - Up to 30 frames per second for resolutions up to 352x288 for Standard System.

- Up to 15 frames per second for resolutions up to 640x480 for Minimum System.
- Color format : I420 & RGB24.
- AVI format.

- Captures stills at all resolutions up to 640x480.
- Attaches to the PC via the Universal Serial Bus (USB) port.
- Small Form Factor.

Since digital camera pictures have better resolution than webcam and people is prone to buy them nowadays, we chose to use also these images as inputs. The reason is it will be easier to implement with digital camera images because of the higher quality with respect a basic webcam. Once the problem is solved for digital camera images the program should be tuned to respond in the same way with a worse resolution.

The features of the camera to be used are the followings:

- CCD resolution: 1/2.7 inch type (3.3 M total pixels)
- Image resolution: 3.1MP(2032x1524 pixels)
- Picture quality: 3.1 MP -best (prints up to 11x14), 2.8 MP -best 3:2 (optimized ratio for 4x6 prints), 2.1 MP -better (small prints), 1.1 MP - good (email)
- Zoom: 3X optical zoom – 5.6-16.8 mm (35 mm equivalent: 37-111 mm), 3.3X digital zoom, 10X total zoom
- Aperture: f/2.7-5.2 (wide), f4.6-8.7 (tele)
- Shutter speed: 1/2 - 1/1400 seconds
- Viewfinder: real image optical viewfinder
- Display: 1.6" (4 cm) TFT indoor/outdoor color display

With the webcam and the digital camera we can deliver different sizes for the book cover pictures. Those sizes will be analyzed in order to see which one gives the best result. For the digital camera in specific will be use the good quality.

4   Region of Interest (ROI) and Segmentation

The region of interest will be defined as the area in the image where the complete book cover appears, vertically oriented. From this area the algorithm will take pixel information to generate the features for the classification. We can assume that more than the 80% percent of the book will be the book itself, and th rest the background. We can also assume that the book cover will be the central part of the picture. The whole image (background and book cover) will be use for recognize the book cover.

5   Features for Recognition

For classification purposes it will be necessary to specify features to extract from every image. The features that will be use for recognition are the Scale Invariant Keypoints (SIFT) by Lowe[9], since they are invariant to scale, rotation and partially invariant

to illumination differences. Another features are Affine Covariant Region Detectors, specially the Harris Affine transformation[7], to be used before the SIFT descriptors. For matching regions between pictures we will experiment with different algorithms like euclidean distance and RANSAC[5].

6 Classification Algorithm

For classification purposes we will use the **K-means** algorithm. This algorithm will help us to group the data available for the recognition (training set) into clusters for a fast retrieval. Each cluster will represent a group of images where they have a high degree of similarity. When a book cover image is captured by the webcam, the algorithm will find the cluster that it corresponds, and then it will be compared with the images in the cluster. Other algorithms will be considered, like SVM for classification.

7 Software

The software to be used for the project will be the affine covariant features implementation from the Visual Geometry Group[12] from the University of Oxford. Also it will be use Matlab and Perl as programming languages, depending of the operations that need to be implemented in the project.

8 Milestones of the Project

The project has been organized in the following milestones:

- **January 9 - 15** Obtain small subset of training data (set for the database and input data). We find to determine what is the best size for training/input images, in order to get a better recognition. Does it make a big impact on the precision? What extension should be use in order to get more information about the images?.

- **January 16 - 29** Generation of image features (keypoints or descriptors). What descriptors can offer a better representation of a book cover? How to compare a book cover and an image that includes the book cover but also a background?. How can we deal with noise?. Implementation.

- **January 30 - February 12** Matching common keypoints in different images. What algorithms are better to accomplish this? What percentage of precision can we obtain? How does it vary when we have less quality (webcam image)?. Implementation.

- **February 13 - 19** Generation of the rest of the training data. Adding more input data. Generation of test sets.

- **February 20 - March 5** Determine algorithms for clustering and retrieval of images. Training of algorithms. What algorithms are better suited for this task?. Dealing with a large database, Do we still have the same performance and precision?. Implementation.

- **March 6 - 17** Retrieval images from database. Build image database. User Interface?. Implementation.

9   Logistical Issues

One of the logistical issues is to obtain the input data from the webcam/camera, that can be quiet time consuming. This is because is necessary to get an image that has the full front cover of the book (trying to avoid rotations and skews) and the fewest background possible.

Respect to the training set and test set, is important to determine which covers need to be taken from Amazon, since most of them will be extracted from Google Print (main source). This step can also be time consuming, but it can be done automatically using a web crawler.

10   Qualifications

Master thesis based on information extraction from the web[11] that involved information retrieval and machine learning techniques. During fall quarter 2005 I studied basic topics on Vision, and acquired some basic background about the area. I've also started the implementation of this project. As a first year Ph.D student I'm very interested on getting into the Vision Learning area, specially in digital libraries.

References

[1] A.C. Berg, T.L. Berg, and J. Malik, Shape Matching and Object Recognition using Low Distortion Correspondence. CVPR 2005.

[2] Amazon http://www.amazon.com.

[3] Apple http://www.apple.com/

[4] Delicious Library http://www.delicious-monster.com/

[5] Fischler, M. A. and Bolles, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24, 6 (Jun. 1981), 381-395.

[6] Google Prints Beta http://prints.google.com.

[7] K. Mikolajczyk and C. Schmid, Scale and Affine invariant interest point detectors. In IJCV 1(60):63-86, 2004.

[8] Louka Dlagnekov, Video-based Car Surveillance: License Plate, Make, and Model Recognition, U.C. San Diego (Masters Thesis).

[9] D. Lowe, Distinctive image features from scale invariant keypoints. In IJCV 2(60):91-110, 2004.

[10] Shivani Agarwal, Aatif Awan, and Dan Roth. Learning to detect objects in images via a sparse, part-based representation.IEEE Trans. on Pattern Analysis and Machine Intelligence, 26(11):14751490, 2004.

[11] Subsumer, http://www.subsumer.com.

[12] Visual Geometry Group, http://www.robots.ox.ac.uk/ vgg/.

Book Cover Recognition

Linfeng Yang(linfeng@stanford.edu) Xinyu Shen(xinyus@stanford.edu).

Abstract—Here we developed a MATLAB based Graphical User Interface for people to check the information of desired books in real-time. The GUI allows user to take photos of the book cover. Then it will automatically detect features of the input image based on MSER algorithm, then it will lter out non-text features based on morphological difference between text and non-text regions. In order to further improve the detection result, we implemented a text character alignment algorithm which signicantly impr The purpose of this project is to recognize book covers from images that are taken by a regular digital camera or a webcam. The recognition will be made through image processing techniques, using SIFT descriptors and transformations, that will allow to identify the correct image that represents the cover from a database of covers. The input images will consist of a book cover and a background were the book was placed. The cover recognition project is base on the Delicious Library[4] for Apple Computers[3], which manages media collections via bar code scanning with an iSight camera, and grabs c...