AUTOMATED OBJECT SHAPE MODELLING BY CLUSTERING OF WEB IMAGES

Giuseppe Scardino, Ignazio Infantino
Istituto di Calcolo e Reti ad Alte Prestazioni ICAR-CNR
Viale delle scienze, edificio 11 Parco d'Orleans - 90128 Palermo, Italy

Salvatore Gaglio
Dipartimento di Ingegneria Informatica DINFO
Viale delle scienze, edificio 6 Parco d'Orleans - 90128 Palermo, Italy

Keywords: Visual image search, images clustering, image annotation.

Abstract: The paper deals with the description of a framework to create shape models of an object using images from the web. Results obtained from different image search engines using simple keywords are filtered, and it is possible to select images viewing a single object owning a well-defined contour. In order to have a large set of valid images, the implemented system uses lexical web databases (e.g. WordNet) or free web encyclopedias (e.g. Wikipedia), to get more keywords correlated to the given object. The shapes extracted from selected images are represented by Fourier descriptors, and are grouped by K-means algorithm. Finally, the more representative shapes of main clusters are considered as prototypical contours of the object. Preliminary experimental results are illustrated to show the effectiveness of the proposed approach.

1 INTRODUCTION

In this paper we explore the possibility to create automatically prototypal shapes of an object using knowledge extracted from the web: images are downloaded by image search engines; given the name of object, more keywords (synonyms, hyponyms, or hypernyms, see (Wordnet,)) are selected from lexical databases. Our approach is unsupervised and tries to select automatically a controlled subset of images in order to assure that image processing algorithm output is correct and fast computed. A supervised approach is presented in (Fergus et al., 2005), using probabilistic Latent Semantic Analysis (pLSA), in order to learn object categories from Google.

In other approaches, user interaction is requested, and for example in (Del Bimbo and Pala, 1997) image retrieval is performed by shape similarity given a user-sketched template. Another possibility is to enforce low level processing by using learning algorithms to recall or classify content of images: for example in (Tieu and Viola, 2004) a very large set of selective features are used and the system learns key features trough given queries. The previous cited references deal with image retrieval, but there are other relevant studies dealing with annotation and clustering: an interesting approach considering colors and textures to annotate images in real time is illustrated in (Jia and Wang, 2003); (Zinger et al., 2006) combines face detection algorithm and color segmentation to cluster and classify images. In this work, we try to use shape as principal feature without a-priori knowledge to detect and to recognize objects in image.

2 FRAMEWORK DESCRIPTION

The proposed framework named POW (Prototypes of Objects from the Web) allows to build shape models of simple objects using *web knowledge*. Given the name of an object, the main steps that characterize the process (see figure 1) are the:

- searching other possible keywords strictly related to given name (synonyms, hyponyms, or hypernyms) using lexical web databases or web encyclopedias;
- collecting images from web using image search

Scardino G., Infantino I. and Gaglio S. (2008).
AUTOMATED OBJECT SHAPE MODELLING BY CLUSTERING OF WEB IMAGES.
In *Proceedings of the Third International Conference on Computer Vision Theory and Applications*, pages 252-255
DOI: 10.5220/0001075002520255

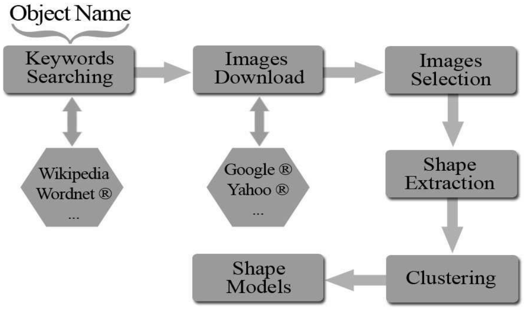

Figure 1: POW (Prototypes of Objects from the Web) - Framework Model: given an object name, the aim is to automatically create a set of shape models of it.

engines using combinations of available keywords;

- creating a set of valid images (deleting duplicates, images too small, and so on);
- collecting closed contours by edge detection and linking of pixels;
- clustering shapes in order to find relevant prototypes;
- validation of prototype searching partially occluded shapes in images previously not considered.

2.1 Populating the Dataset

Popular image search engines (Google® images, Yahoo® images, and so on) use textual information to find images on the web. When a simple word like *chair* is given as search keyword, image links returned include a lot of files not relevant to find a possible shape of this object. To make more robust this critical phase, it is possible to associate to first simple search some more restrictive queries using different combinations of keywords related to *chair*. Merging the results of this multiple searches, we have empirically observed that obtained dataset is optimal for our aims: several images show a single chair and have a background that allows to extract easily object contour. Where additional keywords could be found? Also in this case, we gain this information from the web using lexical web databases or web encyclopedias: the word *chair* is given as keyword to Wordnet® (lexical word database) (Wordnet,) or Wikipedia® (on-line encyclopedia) (Wikipedia,). We have a list of hyponyms by Wordnet, and an html page by Wikipedia that is processed to extract relevant words.

In order to extract valid shapes of the given object and to speed up the computation, it needs to select images that owns the following properties

- acceptable image size: small images are deleted because the possible shape to find is not relevant, and large images are deleted because image processing algorithms might be computationally infeasible;
- uniqueness: it's created a dynamic list with the MD5 signatures of images (Rivest, 1992), so when a new image shows a signature present in the list will be not considered;
- its link to download is valid;
- uniform background: gray level histogram of a small area along the perimeter of the image is calculated (3∼7 pixel of size), and if it cover a limited range of levels make available the image to further elaboration. We indicate this interval of gray levels as R_P.

In this way, the dataset is constituted by images that could be easily segmented by thresholding, and that usually show one o more objects with well-defined shape.

The uniform background permits to characterize an object to create a prototype not influenced by other objects. In successive step the prototype will be used to recognize the object in a generic image.

2.2 Shape Detection and Description

After this first selection of images, we have to find closed contours that define shapes. Different solutions have been tested, and the more robust is resulted the application of following steps

1. choosing a threshold to have a binary image: we select the local minima using only pixels on image border that we suppose is related to background and applying a mean filter;
2. applying an edge detector;
3. searching closed contours: an arbitrary edge point is chosen, and we start to search edges that form a closed path.

The last phase is quite complex and needs to explain some details:

- a dynamic list L_C of coordinates is created, adding new edge points until it is possible or the starting pixel is reached; loops are avoided because new point will be considered if it is not in the list;
- the max hole admissible is dependent from image size (∼20 % of minimum dimension);
- the starting point is the pixel of edge image nearest left upper corner; if a closed contour is not found or the resulting contour is too short (minor of a given threshold), pixels recorded in contour

list L_C are deleted in the edge image and step 3 is repeated;

- algorithm to find a closed path has a max fixed number of iterations in order to limit computational time; when time out is reached the same search procedure is applied to a simplified edge image; this edge map considers only external edge points, i.e. with minor distance from image borders;
- when a closed contour is found, points in list L_C are marked in the edge image with an incremental integer label, and step 3 is repeated in order to find other closed contours until there are unmarked points in the edge image.

After that coordinates of the points of a contour are referred to centroids of shape, Fourier descriptors are used to describe it (see for example (Zhang and Lu, 2002), or (Lee and Long, 2003)). In particular only the first 48 components are stored in order to exclude details or noise from the shape description. This description is independent of object scale and orientation. Moreover, when a metrics is applied in order to evaluate similarity between shapes, it is also possible to recognize specular shapes considering in reverse order elements of a descriptor vectors.

2.3 Object Prototypes by Clustering

Collected shapes can be grouped using a clustering algorithm (see for example (Oliver et al., 2006)) based on similarity of Fourier descriptors (using Euclidean distance). By default the number of clusters is equal to number of keywords provided by Wordnet and Wikipedia. To validate them we calculate the mean distance of objects from cluster center and verifying if a label (combination of keywords, see section 2.1) is predominant. We have use k-Means algorithm choosing at random the initial positions of cluster centers. In our experiment we have used a number of clusters equal to the number of keywords collected from Wikipedia and from Wordnet. We have limited the number of keywords to the maximum of 10 (if they are available).

Some results are reported in figures 3: images are placed in Cartesian space using the second and the third component of Fourier descriptors (to have an approximate idea of similarity), and bars of different colors point to the various clusters. Isolated images, or clusters with few shapes are automatically removed. Figure 2 reported explicative examples of clustering results for *screwdriver* and *chair* (see details in the following section). The objects nearest cluster centers are considered prototypal shapes of the given object.

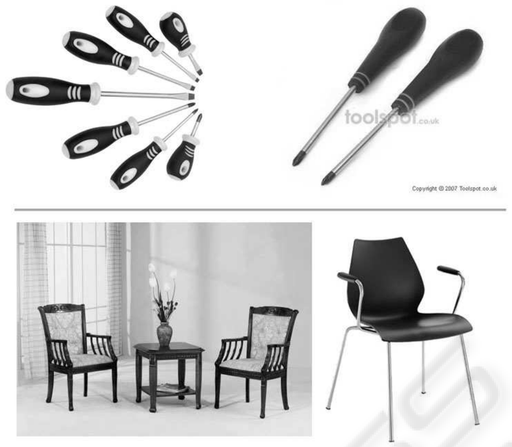

Figure 2: The shape descriptors are based on Fourier descriptors, and they are independent of scale, orientation, and mirroring by a suitably normalization. The figure reports two examples of objects (*screwdriver.* and *chair*) that have shapes recognized as similar.

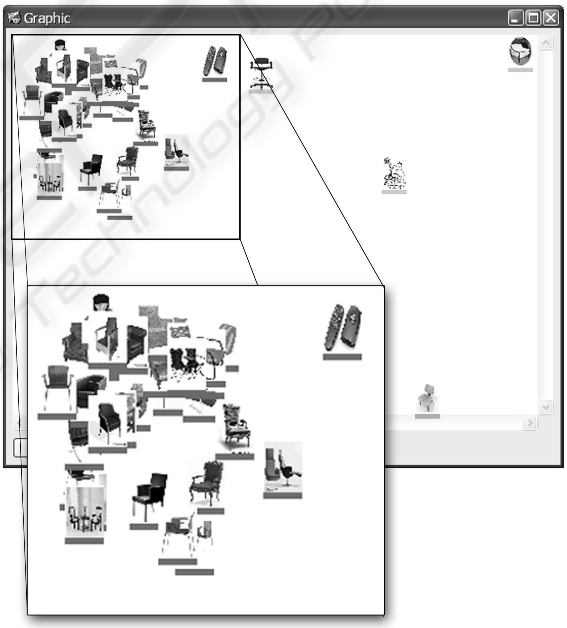

Figure 3: Visualization of one example of clustering related to the word *chair*: images are placed in Cartesian space using the second and the third component of Fourier descriptors. Colored bars refer to different detected clusters.

3 EXPERIMENTAL RESULTS

A complete system based on the proposed framework has been implemented, allowing us to perform experimentations (the beta version is available to download at (POW,)). In order to explain the results, we use a subset of images, this example reported in figure 3

Table 1: Performance evaluations.

	Collecting images	Selection Valid Images		Clustering			
				False negative		False positive	
	Downloads	N.	%	N.	%	N.	%
Bowl	1596	140	9%	8	6%	25	18%
Candle	2032	139	7%	11	8%	32	23%
Chair	4862	602	12%	25	4%	40	7%
Desk	3185	344	11%	40	12%	55	16%
Door	3754	288	8%	00	3%	100	35%
Fork	2264	168	7%	12	7%	20	12%
Glass	4828	396	8%	250	63%	8	2%
Hammer	3561	422	12%	60	14%	55	13%
Knife	4506	551	12%	32	6%	66	12%
Lamp	4773	479	10%	30	6%	16	3%
Pen	3495	460	13%	28	6%	24	5%
Spoon	3699	150	4%	14	9%	22	15%
Sunglasses	962	190	20%	8	4%	25	13%
Torch	1585	112	7%	9	8%	36	32%
Watch	4050	510	13%	44	9%	45	9%
Total (%)	**50324**	**5116**	**10%**	**590**	**12%**	**607**	**12%**

is related to images of word *chair* using different hyponyms from WordNet. In this case the list of first five keywords used for word "*chair*" is: "armchair", "barber", "longue", "chaise", "daybed". The filtering phase individuates 63 (of 602) images that have a detectable shape, and 2 clusters are validated as source of shape models: they have a sufficient number of images ($\geq$ 10), and a mean error under a given threshold. The effectiveness of the approach could be highlighted by some specific examples: in bottom part of figure 2 we see two images of chairs grouped in the same cluster that are impossible to correlate if we consider texture, color, or other image features different from shape; the upper part reports images of screwdriver that demonstrate the independence from scale, orientation, and mirroring.

Results of an extensive experiment are reported in table 1, using 16 different words of common objects. The table shows the number of images downloaded for each word (column 2), the number of valid images to create prototypes and percentage with respect to downloaded images (column 3 and 4), and the performance of clustering (last 4 columns): absolute number and percentage with respect to the number of valid images of images erroneously excluded from relevant clusters, and absolute number and percentage of image of object wrongly included in some clusters. In general results could be considered positive, even if some words are intrinsically difficult to manage for our aims: *glass* images report very different typologies of objects and in this case an interaction with user could be necessary.

Future works will deal with integration in the framework of other visual features (texture and color), in order to have better results. Moreover, it is interesting to explore the possibility to defines categories (or typologies) of the same object using keywords, and to find or define some simple relation among them based on visual features.

ACKNOWLEDGEMENTS

The work of this paper was partially supported by project POR Sicilia 2000-2006,1999/IT.16.1.PO.011/ 3.13/7.2.4/342.

REFERENCES

Del Bimbo, A. and Pala, P. (1997). Visual image retrieval by elastic matching of user sketches. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19 (no. 2), pp. 121-132.

Fergus, R., Fei-Fei, L., Perona, P., and Zisserman, A. (2005). Learning object categories from google's image search. *ICCV*, pages 1816–1823.

Jia, L. and Wang, J. Z. (2003). Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE transaction on pattern analysis and machine intelligence*, vol 25, no. 9.

Lee, D. J. Antani, S. and Long, L. R. (2003). Similarity measurement using polygon curve representation and fourier descriptors for shape-based vertebral image retrieval. *Proceedings of IS&T/SPIE Medical Imaging 2003: Image Processing*, vol. SPIE 5032, pp. 1283-1291.

Oliver, A., Munoz, X., Batlle, J., Pacheco, L., and Freixenet, J. (2006). Improving clustering algorithms for image segmentation using contour and region information. *Automation, Quality and Testing, Robotics, 2006 IEEE Intl. Conf.*, pages 315–320.

POW. Download page:. *http://www.pa.icar.cnr.it/ infantino/demo/*.

Rivest, R. L. (1992). The md5 message digest algorithm. *In: Internet, RFC 1321*.

Tieu, K. and Viola, P. (2004). Boosting image retrieval. *Intl. Journal of Computer Vision*, pages vol. 56(1/2), pp. 1736.

Wikipedia. Home page. *http://wikipedia.org*.

Wordnet. Home page. *http://wordnet.princeton.edu/*.

Zhang, D. and Lu, G. (2002). Shape-based image retrieval using generic fourier descriptor. *Signal Processing: Image Communication*, vol.17, no. 10, pp. 825-848.

Zinger, S., Millet, C., Mathieu, B., Grefenstette, G., Hede, P., and Moellic, P. A. (2006). Clustering and semantically filtering web images to create a large-scale image ontology. *Proc. Of IS-T/SPIE 18th Symposium Electronic Imaging*.