HIGHLY ACCURATE VIDEO OBJECT IDENTIFICATION UTILIZING HINT INFORMATION

Liang Peng[1], Yimin Yang[2], Xiaojun Qi[1], and Haohong Wang[3]

[1]*Department of Computer Science, Utah State University*
[2]*School of Computing and Information Sciences, Florida International University*
[3]*TCL Research America*
liang.peng@aggiemail.usu.edu, yyang010@cs.fiu.edu, Xiaojun.Qi@usu.edu, haohong.wang@tcl.com

Abstract

We propose a hint-information based object identification system for video to significantly improve the object recognition accuracy of the conventional object identification system. To this end, we first formulate a novel cost function to ensure good local representation and good content variation coverage of candidate key frames. We then apply dynamic programming on the cost function to extract key frames from the input video to summarize and represent the whole video. Finally, we recognize the objects in the key frames using the learned model on the conventional knowledge database (i.e., training images) and use these labeled recognized objects as hint information to refine the knowledge database. The good representativeness of hint information alleviates large variations between training and testing images and therefore significantly improves the object recognition performance. As a proof of concept, we use face identification as an example to demonstrate the effectiveness of the proposed hint-information based object identification system. Our extensive experimental results on three types of movies demonstrate the important role of hint information in the proposed system and the excellent performance of the proposed system when compared to the conventional object identification system without using hint information.

Index Terms— Object identification, object detection, object recognition, hint information, key frames

1. INTRODUCTION

Object identification in video is a problem in computer vision that targets at locating and identifying objects (i.e., giving the exact identity) in a video by learning from a given set of images that contain the objects with known identities. Due to the dramatic growth of video-capturing devices and available online video content, video object identification has been attracting a lot of attention in computer vision field. Especially, object identification in video has been driven by its huge potential in developing applications in many domains including video surveillance security, augmented reality, automatic video tagging, medical analysis, quality control, and video-lecture assessment [1] [2]. Even though object identification is a relatively easy task for human brains, it is challenging for computers due to the illumination variation, pose changes, and occlusion among objects of the same kind.

Depending on the scope of the term "object", some research addresses a certain category of objects such as face or car for the recognition task; some other work focuses on identifying the object across different categories. Regardless of the types of work, object identification typically involves object detection and object recognition steps. For either detection or recognition, existing work in this domain generally consists of two stages [3][4]: learning and recognition phases. In the learning stage, a database of static images (i.e., knowledge database) containing different objects is used as training data. Based on the specific category of objects, features with high discriminative power are extracted. These features are further combined with a certain learning schema to develop a model. In the recognition stage, the new unknown objects are detected and classified as a certain object by the learned model.

Most research in this domain has been focusing on finding more discriminative features and more effective learning schema to improve the recognition accuracy [5]. However, a common difficulty in object recognition in video, especially in unconstrained video, is that the static database used for training usually contains objects that differ greatly from the objects in testing images or videos in terms of orientation, illuminance, occlusion, etc. Video sequence contains a large number of frames which include intrinsic spatio-temporal information that could be used to extract hint information to help object identification. Although there is some existing work using clustering-based method for selecting representative exemplars from a large corpus of static images [6], effectively extracting useful and compact information from video as a hint to help with object identification is a challenging research problem which has not been deeply explored.

We propose a novel approach addressing object identification in video with the following novelties: 1) Formulate a new cost function to ensure two properties (i.e., good local representation and good content variation coverage) of candidate key frames. 2) Apply dynamic programming on the cost function to extract key frames to summarize and represent the input video. 3) Generate hint information to refine the knowledge database by using a learned model on the training images in the conventional knowledge database. 4) Incorporate object detection, hint information generation, and object recognition in the final system to significant boost the object recognition performance in video. The rest of the article is organized as follows: we describe the proposed object identification framework in section 2; the

experiments and results are presented in section 3. Finally the work is concluded in section 4.

2. PROPOSED OBJECT IDENTIFICATION FRAMEWORK

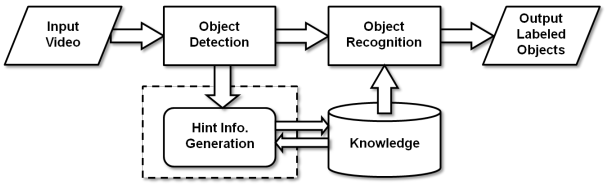

Fig. 1. Workflow of proposed object identification system

The proposed object identification framework is shown in Fig. 1. It incorporates a new component, hint information generation (as shown in box of dashed line), into a typical object identification framework. In a typical object identification framework (shown in Fig. 1 without the box of dashed line), an object detection module is applied to the input video and then an object recognition module (possibly including tracking) is applied to the detected objects to identify objects in a certain portion of frames of input video by using a learning model constructed from a knowledge database of labeled objects (i.e., training data). In the proposed framework, hint information is generated to build a more effective knowledge database to include highly informative and representative objects as the training data. This refined knowledge database contains training images which have an overall less variation with the candidate objects in the testing video. As a result, the proposed framework significantly improves the object identification performance when compared to the typical object identification framework. It consists of three major components: object detection, hint information generation, and object recognition. In the following subsections, we describe each of these three components in detail.

2.1 Object Detection

Detecting objects in a video is a necessary step before object recognition. Object detection refers to automatically finding the location of objects of interest in a sequence of frames. There are a lot of existing work on object detection in a video [7]. Some focus more on detecting a specific category of objects (e.g., face and car) [8] [9] while some take a more generic approach [10]. The object detection module, as the first component in the proposed framework, aims at detecting the pre-defined objects of interest. Hence, the specific detectors to be used is flexible. For example, if the objects of interest include multiple categories such as face, car, and bag, the detectors with trained models for each category could be used together to detect objects of interest. The input video is processed frame by frame by running the object detection algorithm on each frame. The frames in which the objects of interest may appear are then recorded with the location (e.g., bounding box) of each object in each of these frames. Taking face detection as an example, we could use the Viola & Jones face detection algorithm [8] to detect faces in each frame so we can select the frames that contain detected faces for further processing. Similarly, other objects such as bag and car can be detected using the popular object detection algorithm proposed in [11]. Fig. 2 illustrates detection results for three specific objects including face, bag, and car by applying their respective detection algorithms [8, 11]. The three detected objects are highlighted in green bounding boxes. It should be noted that these object detection algorithms can be easily replaced by the state-of-the-art detection algorithms to achieve better detection results.

(a) Face and bag (b) Car

Fig. 2. Object detection examples

2.2 Hint Information Generation

Hint information generation refers to the generation of informative and representative objects to refine the knowledge database and further improve the recognition performance. To address the large variation issue between images in the knowledge database and the testing video, which is the common challenge of object recognition in an unconstrained video, we propose to use hint information as training data to refine the knowledge database. This refined knowledge database is further used by the learned model for the recognition task. To this end, we first filter out the frames without objects of interest based on the object detection results. We then apply dynamic programming to obtain a small subset of frames (i.e., key frames) from the remaining input video to best summarize and represent the whole video. Finally, we identify objects in key frames by applying the learned model on the conventional knowledge database. These identified objects are informative and representative objects from video and therefore reduce the variations between training and testing data in video. They are the hint information in the proposed framework and are used to refine the knowledge database. In summary, we initially perform the object recognition task on key frames to generate hint information by applying the learned model on the conventional knowledge database. We then use labeled hint information as training data to refine the knowledge database to perform object recognition on the whole video. Here, key frames extraction is the most important step since good hint information can only be generated from representative frames best summarizing the input video. In the next two subsections, we explain the problem formulation of the key frames extraction and its corresponding solution. In the final subsection, we explain the refinement of the knowledge database.

2.2.1 Problem Formulation

The key frames should possess two properties: 1) Good local representation: Each selected key frame has good representativeness for its neighboring frames; and 2) Good content variation coverage: The selected frames cover large content variations from the whole video. Based on these two properties, we formulate the key frames extraction problem as a cost optimization problem. Let N represent the total number

of frames in a video to be summarized (in our framework, it is the total number of frames which contain detected objects), and $\{F_i\}$, $(i = 1, ..., N)$ represent the set of frames containing the objects of interest. The goal is to select M frames $\{a_i\}(i = 1, ..., M)$ from $\{F_i\}$ which are the best representative frames to summarize the video.

Good local representation means that each selected frame has large visual similarity with its neighboring frames (i.e., each selected frame is similar enough to represent its neighboring frames from the original video). Let H_i stand for the 1-D feature vector (the choice of this specific feature vector depends on the type of objects to be recognized) of the ith frame, F_i. Let Sim represent a function that computes the similarity between two 1-D vectors. The local representation of ith frame F_i is defined as:

$$P(i) = \begin{cases} Sim(H_i, H_{i+1}) & \text{if } i = 1 \\ \frac{Sim(H_{i-1}, H_i) + Sim(H_i, H_{i+1})}{2} & \text{if } 1 < i < N \\ Sim(H_{i-1}, H_i) & \text{if } i = N \end{cases} \quad (1)$$

That is, the local representation of F_i is computed by the average similarity between its previous frame F_{i-1} and next frame F_{i+1} (except for the first and last frames whose local representation is computed by its similarity with the second frame and its similarity with the second to the last frame, respectively).

A good content variation coverage can be interpreted as that consecutively selected frames have a large dissimilarity. Here, we define the similarity of the two key frames containing the objects of interest by:

$$Q(k, j) = Sim(H_k, H_j), \quad 1 \le k \le N,\ 1 \le j \le N,\ k \ne j, \quad (2)$$

where H_k and H_j are the 1-D feature vectors of two selected frames from F_k and F_j, respectively.

A good summary of the video requires larger $\sum_{i=1}^{M} P(a_i)$ (i.e., better local representation), and smaller $\sum_{i=2}^{M} Q(a_{i-1}, a_i)$ (i.e., better content variation coverage). Then we create a total cost function C:

$$C(a_1, a_2, ..., a_M) = \sum_{i=1}^{M} \alpha[1 - P(a_i)] + \sum_{i=2}^{M} (1 - \alpha) Q(a_{i-1}, a_i), \quad (3)$$

where α is a weighting factor with its value between $[0, 1]$. This total cost function consists of cost for local representation based on a single frame and content variation coverage based on a pair of consecutive key frames.

2.2.2 Solution to Key Frames Extraction

The goal is to find the solution set $(a_1^*, \cdots, a_M^*)$ to minimize the cost function (3), which can be solved by dynamic programming [12]. Let $\Omega_i[a_i]$ denote the cost based on the optimal selection of the first i frames, i.e., $\Omega_i[a_i] =$ Minimize $C(a_1, \cdots, a_i)$. For selecting the $1st$ key frame, the cost function only involves cost for local representation since the content variation coverage cost is zero. Hence, we have

$$\Omega_1[a_1] = \alpha(1 - P(a_1)) \quad (4)$$

as the base case. In addition, by definition of $\Omega_i[a_i]$, we have

$$\Omega_i[a_i] = \Omega_{i-1}[a_{i-1}] + \alpha(1 - P(a_i)) + (1 - \alpha) Q(a_{i-1}, a_i) \quad (5)$$

which shows that the selection of the next key frame index a_i is independent of the selection of previous frames for a given cost function. Once Ω_i is computed, the optimal solution to the whole problem can be obtained by taking:

$$a_M^* = \underset{a_M}{\operatorname{argmin}}\, \Omega_M[a_M] \quad (6)$$

and tracking back in order of decreasing i until the base case (4) is satisfied. In other words,

$$a_{i-1}^* = \underset{a_{i-1}}{\operatorname{argmin}} \{\Omega_{i-1}[a_{i-1}] + \alpha(1 - P(a_i^*)) + (1 - \alpha) Q(a_{i-1}, a_i^*)\} \quad (7)$$

The computational complexity of this dynamic programming approach is $O(MN^2)$.

2.2.3 Knowledge Database Refinement

After key frames extraction, we crop out the objects from key frames and use the conventional knowledge database (i.e., the exiting objects with known labels which is not extracted from video) as training data to perform the classification on these objects. For the face example, we use Local Binary Pattern (LBP) [13] as the feature vector and apply K-Nearest-Neighbor (KNN) [14] as the classifier to recognize (i.e., label) all cropped faces from key frames. These labeled data are used as hint information to perform recognition. The accuracy in the object recognition step can be further improved if the misclassified labels in the training data are corrected. In this way, the knowledge database is refined with correct hint information (i.e., correct labeled informative and representative objects).

2.3 Object Recognition

Object recognition typically refers to using a set of static images (training data) to classify each of the detected objects into one single class. For example, face recognition typically means using a set of training images that contain different faces to classify each detected face in a new set of images into a pre-defined face [15] if their similarity passes a certain threshold. Specifically, using face as an example, we first extract the LBP feature vector for each face of each frame in the video. We then compare the LBP vector of each face with the LBP vectors of all faces in training data (i.e., refined knowledge database) by a certain similarity measure (e.g. Euclidean distance) to find the nearest neighbor. If this similarity passes a pre-defined threshold, we would assign this face with the label of the nearest face. Otherwise, we classify it as unknown. Similar to the object detection step, we can choose the state-of-the-art methods for each category of object for recognition.

3. EXPERIMENTAL RESULTS

To evaluate the effectiveness of the proposed object identification system, we divide videos into three types based on the level of movements. These three types are videos with frequent

movements (Type I), videos with intermediate movements (Type II), and videos with infrequent movements (Type III). We conduct a comparative experiment using face recognition as an example incorporating hint information or without incorporating hint information.

3.1 Dataset and Evaluation Criteria

In the experiment, three 20-minute video clips with one from each type are selected. The movie "The Warring States" (movie one) belongs to Type I; The movie "Crouching Tiger Hidden Dragon" (movie two) belongs to type II; and the movie "Cell Phone" (movie three) belongs to Type III. Three to four main characters are chosen from each movie for face recognition. In the "with hint" setting, a set of training faces from each movie are automatically selected serving as hint information; while in the "without hint" setting, the same number of training faces for each person are downloaded from the Google image search results based on each actor's name. Some statistics of the movie clips are shown in Table 1. It should be mentioned that a huge selection of training images for each actor can be added to the conventional knowledge database without hint information to achieve comparable performance as the one achieved by the proposed system, where a significantly smaller knowledge database with hint information is employed.

Table 1. Movie statistics

Movie name	**Total # of frames**	**# of main characters**	**# of frames with faces**	**# of key frames**	**# of faces for training**
Movie one	30295	4	2734	77	85
Movie two	30532	3	3755	105	116
Movie three	32508	3	1386	39	51

The evaluation criteria are the Precision (Pre.), Recall (Rec.) and F1 score, which are defined as follows.

$$\text{Precision} = \frac{\#\ of\ correctly\ recognized\ faces}{\#\ of\ recognized\ faces}, \quad (8)$$

$$\text{Recall} = \frac{\#\ of\ correctly\ recognized\ faces}{total\#\ of\ faces}, \quad (9)$$

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (10)$$

3.2 Quantitative Analysis

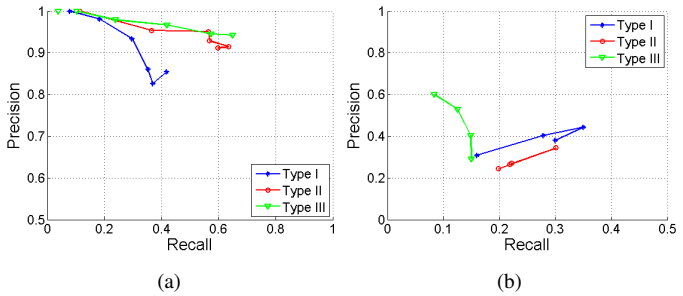

Fig. 3. Precision-Recall curve obtained with (a) and without (b) hint information

The experiment is conducted using KNN (K = 1) classifier with the LBP feature vector under different thresholds. Based on the difference in training data with and without hint information (i.e., the difference in the refined and conventional knowledge databases) as well as empirically results of precision-recall under a large set of thresholds, we make the plots based on the range of thresholds to which precision-recall changes are most sensitive. Specifically, six optimal thresholds (i.e., 80, 100, 120, 140, 160, and 180) are chosen for the proposed system with hint information, while four optimal thresholds (i.e., 180, 200, 220, and 240) are chose for the system without hint information. A random sample of size 200 is taken under each setting to compute precision and recall. The precision-recall curves with and without hint information are shown in Fig. 3(a) and Fig. 3(b), respectively. The precision, recall, and F1 together with each used threshold is shown in Table 2. From Fig. 3(a) and Fig. 3(b), we clearly see the precision-recall curve for all three movies obtained with hint information is significantly superior to the one obtained without hint information. Using hint information, the precision rate could be above 0.9 while still preserving the recall rate over 0.6. Without using hint information, the highest precision is 0.6 and the recall is around 0.1. Comparing three types of movies, the best recognition performance shown in green color is for movie three (Type III). In contrast, the relatively poor performance for the other two movie types is due to the large number of scene changes and frequent movements.

3.3 Robustness Analysis

To evaluate the robustness in correctly recognizing different faces, we illustrate some face recognition results of our proposed system (utilizing the hint information) in Fig. 4. The recognized faces are highlighted in red rectangles, on top of which are the names of the characters. It is worth mentioning that we are only interested in the main characters in a movie. In other words, the irrelevant and unimportant faces are ignored and therefore are not identified. To facilitate the discussion, a sample image for each main character in each of the three movies is shown in Fig. 5. These sample images are selected from the refined knowledge database. Fig. 4 clearly demonstrates the robustness of the proposed system since characters in different consumes, poses, and scales are correctly recognized and the characters with similar looking are correctly distinguished. For example, faces in Fig. 4(a) and Fig. 4(b) belong to the same character "Xishan Jin" in movie one. Fig. 4(c) has the same character as in Fig. 5(e) with a relatively small scale under low luminance. As for Fig. 4(d), there is a similar face (but not the same) to the character in Fig. 5(e) and another unimportant face. Hence, both faces in this frame are not labeled. Finally, Fig. 4(e) contains two recognized faces in the same frame and Fig. 4(f) has one main role and the other irrelevant face, which is not labeled.

To summarize, our proposed face recognition system is capable of recognizing faces robustly with variant filming constraints due to the benefits of hint information explored from movies.

4. CONCLUSIONS

We propose a novel framework that can identify objects in video by generating hint information to refine the knowledge database. This knowledge database contains labeled informative and representative objects as training data, which also alleviate

Table 2. Performance evaluation results

Movie Type		Thresholds (With hint, Without hint)																	
		(80, -)			(100, 180)			(120, 200)			(140, 220)			(160, 240)			(180, -)		
		Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1
Type I	(With hint)	1.000	0.115	0.206	0.954	0.365	0.528	0.951	0.564	0.708	0.929	0.569	0.706	0.915	0.635	0.750	0.912	0.598	0.722
	(Without hint)				0.270	0.222	0.244	0.344	0.301	0.321	0.265	0.218	0.239	0.244	0.198	0.219			
Type II	(With hint)	1.000	0.079	0.146	0.981	0.183	0.308	0.934	0.297	0.451	0.861	0.352	0.500	0.826	0.370	0.511	0.854	0.418	0.561
	(Without hint)				0.308	0.160	0.211	0.403	0.278	0.329	0.443	0.350	0.391	0.380	0.300	0.335			
Type III	(With hint)	1.000	0.038	0.073	1.000	0.103	0.187	0.979	0.240	0.385	0.967	0.420	0.586	0.945	0.574	0.714	0.942	0.650	0.769
	(Without hint)				0.600	0.084	0.147	0.530	0.125	0.202	0.402	0.149	0.217	0.290	0.150	0.198			

(a) (b) (c) (d) (e) (f)

Fig. 4. Examples of face recognition results with hint information

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 5. Main characters in three movies: (a), (b), (c), and (d) show the four main characters in movie one; (e), (f), and (g) show the three main characters in movie two; (h), (i), and (j) show the three main characters in movie three.

large variations between training and testing images. As a result, a more effective learning model can be built to significantly improve the object recognition performance. The experiments on three video clips show that the proposed system with hint information dramatically outperforms the system without hint information.

5. REFERENCES

[1] Jun Rekimoto, "Matrix: A realtime object identification and registration method for augmented reality," in *Proceedings of 3rd Asian Pacific Computer and Human Interaction*. IEEE, 1998, pp. 63–68.

[2] Richard Stephan Szeliski, Edward Hsiao, Sudipta Narayan Sinha, Krishnan Ramnath, Charles Lawrence Zitnick III, and Simon John Baker, "Object identification using 3-d curve matching," June 20 2013, US Patent 20,130,156,329.

[3] Dilip K Prasad, "Survey of the problem of object detection in real images," *International Journal of Image Processing (IJIP)*, vol. 6, no. 6, pp. 441, 2012.

[4] Rong Yan, Jian Zhang, Jie Yang, and Alexander G Hauptmann, "A discriminative learning framework with pairwise constraints for video object classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 578–593, 2006.

[5] Tao Meng and Mei-Ling Shyu, "Concept-concept association information integration and multi-model collaboration for multimedia semantic concept detection," *Information Systems Frontiers*, pp. 1–13, 2013.

[6] Yimin Yang and Shu-Ching Chen, "Disaster image filtering and summarization based on multi-layered affinity propagation," in *IEEE International Symposium on Multimedia (ISM)*. IEEE, 2012, pp. 100–103.

[7] Rainer Lienhart and Jochen Maydt, "An extended set of haar-like features for rapid object detection," in *International Conference on Image Processing (ICIP)*. IEEE, 2002, vol. 1, pp. 900–903.

[8] Paul Viola and Michael J Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[9] Cheng-Hao Kuo and Ramakant Nevatia, "Robust multi-view car detection using unsupervised sub-categorization," in *Workshop on Applications of Computer Vision (WACV)*. IEEE, 2009, pp. 1–8.

[10] Hae Jong Seo and Peyman Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, 2010.

[11] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[12] Amir A Amini, Terry E Weymouth, and Ramesh C Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 855–867, 1990.

[13] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[14] Daniel T Larose, "k-nearest neighbor algorithm," *Discovering Knowledge in Data: An Introduction to Data Mining*, pp. 90–106, 2005.

[15] Stan Z Li and Anil K Jain, *Handbook of face recognition*, springer, 2011.