

ROBUST MOVIE CHARACTER IDENTIFICATION AND THE SENSITIVITY ANALYSIS

Jitao Sang^{1,2}, Chao Liang^{1,2}, Changsheng Xu^{1,2}, Jian Cheng^{1,2}

¹National Lab of Pattern Recognition, Institute of Automation, CAS, Beijing 100190, China

²China-Singapore Institute of Digital Media, Singapore, 119615, Singapore

{jtsang, cliang, csxu, jcheng}@nlpr.ia.ac.cn

ABSTRACT

Automatic face identification of characters in movies has drawn significant research interests and led to various applications. It is a challenging problem due to the huge variation in the appearance of each character. Although existing methods demonstrate promising results in clean environment, the performances are limited in complex movie scenes due to the noises generated during the face tracking and face clustering process. In this paper we present a robust character identification approach by incorporating a noise insensitive relationship representation and a graph matching algorithm. Beyond existing character identification approaches, we further perform explicit sensitivity analysis on character identification by introducing two types of simulated noises. Experiments validate the advantage of the proposed method.

Index Terms— Character Identification, Sensitivity Analysis

1. INTRODUCTION

Automatic character identification in movies is essential for semantic movie analysis such as movie indexing, summarization and retrieval. Character identification, though very intuitive to humans, is a tremendously challenging task in computer vision. This is due to the introduced noises of huge variation in appearance of characters, such as scale, pose, illumination, expression and wearing. The objective of this work is to explicitly consider the unavoidable noises and analyze the noise impact.

We build on our previous work [1] and present a novel approach for robust character identification. Regarding the fact that characters may show various appearances, the representation of character is often affected by the noise introduced by face tracking, face clustering and scene segmentation. We observed in our investigations that some statistic properties of characters are preserved in spite of these noises. Hence, we investigate a method for character relationship representation and name-face matching which can accommodate a certain noise. Although extensive research efforts have been concentrated on character identification and many applications have been proposed, there have been rather few efforts directed at

the robustness analysis of character identification versus inevitable noises. Here we aim to fill this gap by introducing two types of simulated noises and performing an in-depth sensitivity analysis.

1.1. Related Work

The crux of the character identification problem is to exploit the relations between videos and the associated texts in order to label the faces of characters with names. Name-it [2] is the first name-face association system proposed for news video, which is based on the co-occurrence between the detected faces and names extracted from the transcript. A face is labeled with a name which frequently co-occurs with it. Yang et al. [3] employed the closed caption and speech transcript to build models predicting the probability that a name in the text matches to a face on the video frame. In [4], they improved their method by utilizing multiple instance learning for partial labeled faces to reduce the data collecting effort. Early name-face association methods showed promising results in news videos, where candidate names for the faces are available from the simultaneous appearing captions or local transcripts. However, in TV and movies, the names of characters seldom directly show in the subtitle, and script containing character names has no time stamps to align to the video.

To cope with this problem, Everingham et al. [5] [6] proposed to align the film script and the subtitle to generate time stamped name annotation. Based on that, they learned character specific classifiers from video and extended the coverage of the method in [7]. In their work, the subtitle text and time-stamps information were extracted by OCR, which required extra computation cost on spelling error correction. Instead of OCR and local matching, Zhang proposed a global matching method utilizing the movie script [1]. Two graphs, name affinity graph computed from co-occurrence between character names in the script, and face affinity graph computed from co-occurrence between faces in the video, are first constructed. Then, character identification is formulated as the problem of finding optimal vertex to vertex matching between two graphs. The author concluded that less information source is involved in their method (without subtitle) and the global matching framework improves the performance.

Various applications based on character identification have been investigated. A character-centered browsing platform was provided in [1], where users use name to query the characters of interest and view the related video clips. Liang et al. [8] presented scenes in film script and shots in video as bag-of-characters after character identification, and then solved the movie scene segmentation task by video/script alignment. Recently Sang [9] employed the movie scene structure and leading characters based on character identification and proposed a character-based movie summarization framework.

Although extensive research investigations have been concentrated on character identification, as far as we know, there have been rather few efforts focused on the analysis of noises to improve the robustness and no work has been performed on sensitivity analysis of character identification versus noises.

1.2. Overview of Our Approach

In a movie, the interactions among characters resemble them into a relationship network. Co-occurrence of names in script and faces in videos can represent such interactions. Based on that, one name affinity graph and one face affinity graph can be built respectively. As noises are unavoidable, the face affinity graph can be seen as a transform from the name affinity graph by affixing noises. Through our investigations on the affected affinity graph, these transforms preserve certain statistic properties of the characters. Hence, in this paper, we utilize the preserved properties and investigate a method for robust character relationship representation (graph construction) and name-face graph matching.

For relationship representation, we propose to represent the character co-occurrence in rank ordinal level [10], scoring the strength of the relationships in a rank order from the weakest to strongest. Rank order data carries no numerical meaning. It follows that it is less sensitive to the noise. The affinity graph presented in [1] is interval measures of the co-occurrence relationship between characters. While continuous measures of the strength of relationship holds complete information, it is highly sensitive to noises.

For name-face graph matching, we employ the error correcting graph matching algorithm. In error correcting graph matching (*ECGM*), the difference between two graphs is measured by edit distance which is defined by a sequence of graph edit operations. According to the noise analysis, we define appropriate graph edit operations and constitute the edit distance function adapted to obtain the improved name-face matching performance.

Sensitivity analysis plays an important role in characterizing the uncertainties associated with a model. To explicitly analyze the algorithm’s sensitivity to noises, two types of noises, coverage noise and intensity noise, are introduced. Based on that, we perform sensitivity analysis by investigat-

Table 1. Example of Affinity Matrix
(a) Name Affinity Matrix

	WIL	SPI	ANN	MAX	BEL
WIL	0.173	0.024	0.129	0.009	0.013
SPI	0.024	0.017	0.007	0.001	0.002
ANN	0.129	0.007	0.144	0	0
MAX	0.009	0.001	0	0.009	0.006
BEL	0.013	0.002	0	0.006	0.011

(b) Face Affinity Matrix

	Face1	Face2	Face3	Face4	Face5
Face1	0.186	0.041	0.147	0.008	0.021
Face2	0.041	0.012	0.005	0.002	0.004
Face3	0.147	0.005	0.157	0	0.003
Face4	0.008	0.002	0	0.005	0.007
Face5	0.021	0.004	0.003	0.007	0.009

Table 2. Example of Rank Ordinal Affinity Matrix Corresponding to Table 1

(a) Name Rank Ordinal Matrix

	WIL	SPI	ANN	MAX	BEL
WIL	5	3	4	1	2
SPI	4	3	3	1	2
ANN	2	1	4	0	0
MAX	3	1	0	1	2
BEL	3	1	0	2	2

(b) Face Rank Ordinal Matrix

	Face1	Face2	Face3	Face4	Face5
Face1	5	3	4	1	2
Face2	4	3	3	1	2
Face3	3	2	4	0	1
Face4	3	1	0	1	2
Face5	4	2	1	3	2

ing the performance of name-face matching with respect to the simulated noises.

Compared with the previous work, the contributions of our work include: 1) A noise-insensitive relationship representation method is introduced to construct the name/face affinity graph. 2) An *ECGM*-based graph matching algorithm with specially designed edit cost function is presented for face-name graph matching. 3) We perform an explicit sensitivity analysis on character identification by introducing two types of simulated noises.

2. ROBUST CHARACTER IDENTIFICATION

In this section we first briefly review our previous work on character identification by global name-face graph matching. Based on investigations of the noises generated during the affinity graph construction process, we construct the name and face affinity graph in rank ordinal level and employ *ECGM* with specially designed edit cost function for name-face match.

2.1. Review of Global Name-Face Matching Framework

In movies, the names of characters seldom directly appear in the subtitle, while the movie script which contains character names has no time information. Hence, the task of character identification can be formulated as a global matching problem between the faces detected from the video and the names extracted from the movie script. Affinity graph is built according to the co-occurrence status among characters, which can be represented as a weighted graph $G = \{V, E\}$ where vertex V denotes the characters and edge E denotes relationships among them. The more scenes where two characters appear together, the closer they are, and the larger the edge weights between them is. In this sense, a name affinity graph from script analysis and a face affinity graph from video analysis can be constructed. Table 1 demonstrates the adjacency matrices corresponding to the name and face affinity graphs from the movie "Noting Hill" (the ground truth is *WIL-Face1*, *SPI-Face2*, *ANN-Face3*, *MAX-Face4*, *BEL-Face5*). All the affinity values are normalized into the interval $[0, 1]$. We can see that some of the face affinity values differ much from the corresponding name affinity values (e.g. $\{WIL, SPI\}$ and $\{Face1, Face2\}$, $\{WIL, BEL\}$ and $\{Face1, Face5\}$) due to the introduced noises. Subsequently, a spectral graph matching algorithm is applied between the name affinity graph and the face affinity graph to find the optimal name-face correspondence. More technical details can be referred to our previous work [1].

2.2. Rank Ordinal Affinity Graph Construction

We observed in our investigations that, in the generated affinity matrix some statistic properties of the characters are relatively stable and insensitive to the noises, such as character A has more affinities with character B than C, character D has never co-occurred with character A, etc. Delighted from this, we assume that while the absolute quantitative affinity values are changeable, the relative affinity relationships between characters (e.g. A is more closer to B than to C) and the qualitative affinity values (e.g. whether D has co-occurred with A) usually remain unchanged. In this paper, we utilize the preserved statistic properties and propose to represent the character co-occurrence in rank order.

We denote the original affinity matrix as $R = \{r_{ij}\}_{n \times n}$, where n is the number of characters. First we look at the cells along the main diagonal (e.g. A co-occur with A, B co-occur with B). We rank the diagonal affinity values r_{ii} in ascending order, then the corresponding diagonal cells \tilde{r}_{ii} in the rank ordinal affinity matrix \tilde{R} is:

$$\tilde{r}_{ii} = I_{r_{ii}} \quad (1)$$

where $I_{r_{ii}}$ is the rank index of original diagonal affinity value r_{ii} . For each row in the original affinity matrix, we first remain the 0-cell unchanged. Then, other than the diagonal cell and 0-cell, we rank the rest affinity values in ascending order

(the rank order of r_{ij} is denoted as $I_{r_{ij}}$). Then for the i^{th} row, the corresponding cells \tilde{r}_{ij} in the i^{th} row of affinity rank ordinal matrix is:

$$\tilde{r}_{ij} = I_{r_{ij}} \quad (2)$$

We illustrate in Table 2 an example of rank ordinal affinity matrices corresponding to the affinity matrices in Table 1. The rank ordinal matrix is not necessarily symmetric and the scales reflect differences in degree of intensity, but not necessarily equal differences. It is shown that although there are major differences between original name and face affinity matrices, the derived rank ordinal affinity matrices are basically the same. A rough conclusion is that the affinity rank ordinal matrix is less sensitive to the noises than the original affinity matrix, which coincides with our assumption. We will further valid the advantage of rank ordinal graph in the experiments.

3. ERROR CORRECTING GRAPH MATCHING

Error correcting graph matching (*ECGM*) is a powerful concept that has various applications in pattern recognition and computer vision. Its application is focused on distorted inputs [11]. In order to measure the similarity of two graphs, graph edit operations are defined, such as the deletion, insertion and substitution of nodes and edges. Each of these operations is further assigned a certain cost. The costs are application dependent and usually reflect the likelihood of graph distortions. The more likely a certain distortion is to occur, the smaller is its cost. Through error correcting graph matching, we can define appropriate graph edit operations according to the noise investigation and design the edit cost function to improve the performance.

For explanation convenience, we provide some notations and definitions taken from [12]. Let \mathcal{L} be a finite alphabet of labels for nodes and edges.

Notation: A graph is a triple $g = (\mathcal{V}, \alpha, \beta)$, where \mathcal{V} is the finite set of nodes, $\alpha : \mathcal{V} \rightarrow \mathcal{L}$ is node labeling function, and $\beta : \mathcal{E} \rightarrow \mathcal{L}$ is edge labeling function.

The set of edges \mathcal{E} is implicitly given by assuming that graphs are fully connected, i.e., $\mathcal{E} = \mathcal{V} \times \mathcal{V}$. Node and edge labels (for weighted graphs, edge label is the weight of the edge) come from the same alphabet for notational convenience.

Definition 1. Let $g_1 = (\mathcal{V}_1, \alpha_1, \beta_1)$ and $g_2 = (\mathcal{V}_2, \alpha_2, \beta_2)$ be two graphs. An error correcting graph matching (*ECGM*) from g_1 to g_2 is a bijective function $f : \hat{\mathcal{V}}_1 \rightarrow \hat{\mathcal{V}}_2$, where $\hat{\mathcal{V}}_1 \subseteq \mathcal{V}_1$ and $\hat{\mathcal{V}}_2 \subseteq \mathcal{V}_2$.

We say that node $x \in \hat{\mathcal{V}}_1$ is substituted by node $y \in \hat{\mathcal{V}}_2$ if $f(x) = y$. If $\alpha_1(x) = \alpha_2(f(x))$, the substitution is called an identical substitution. The cost of identical node or edge substitution is usually assumed to be zero, while the cost of any other edit operation is greater than zero.

Definition 2. The cost of an *ECGM* $f : \hat{\mathcal{V}}_1 \rightarrow \hat{\mathcal{V}}_2$ from graph $g_1 = (\mathcal{V}_1, \alpha_1, \beta_1)$ to $g_2 = (\mathcal{V}_2, \alpha_2, \beta_2)$ is given by

$$\begin{aligned} \gamma(f) = & \sum_{x \in \mathcal{V}_1 - \hat{\mathcal{V}}_1} c_{nd}(x) + \sum_{x \in \mathcal{V}_2 - \hat{\mathcal{V}}_2} c_{ni}(x) \\ & + \sum_{x \in \hat{\mathcal{V}}_1} c_{ns}(x) + \sum_{e \in \hat{\mathcal{E}}_1} c_{es}(e) \end{aligned} \quad (3)$$

where $c_{nd}(x)$ is the cost of deleting a node $x \in \mathcal{V}_1 - \hat{\mathcal{V}}_1$ from g_1 , $c_{ni}(x)$ is the cost of inserting a node $x \in \mathcal{V}_2 - \hat{\mathcal{V}}_2$ in g_2 , $c_{ns}(x)$ is the cost of substituting a node $x \in \hat{\mathcal{V}}_1$ by $f(x) \in \mathcal{V}_2$, and $c_{es}(e)$ is the cost of substituting an edge $e = (x, y) \in \hat{\mathcal{V}}_1 \times \hat{\mathcal{V}}_1$ by $e' = (f(x), f(y)) \in \mathcal{V}_2 \times \mathcal{V}_2$.

Definition 3. Let f be an *ECGM* from g_1 to g_2 and \mathcal{C} a cost function. We will call f an optimal *ECGM* under \mathcal{C} if there is no other *ECGM* f' from g_1 to g_2 with $\gamma_{\mathcal{C}}(f') < \gamma_{\mathcal{C}}(f)$

In our cases, the name and face affinity graph have the same number of nodes. There is no need to search for sub-graph isomorphisms and thus, $|\hat{\mathcal{V}}_1| = |\mathcal{V}_1| = |\hat{\mathcal{V}}_2| = |\mathcal{V}_2|$. Also, as no node deletion or insertion operation is involved, we can directly assign $c_{nd} = c_{ni} = \infty$. According to the investigation on noises, we introduce $c_{edc}(e)$ for the cost of destroying a edge $e \in \hat{\mathcal{V}}_1 \times \hat{\mathcal{V}}_1$ or creating a edge $e \in \mathcal{V}_2 \times \mathcal{V}_2$. The edit operation of destroying a edge means certain cell in the name affinity rank ordinal matrix is nonzero while the corresponding cell in the face affinity rank ordinal matrix is zero. The edit operation of creating a edge means the opposite. We define the cost of an *ECGM* in our name/face affinity rank ordinal graph matching application as:

$$\begin{aligned} \gamma(f) = & \sum_{x \in \mathcal{V}_1 - \hat{\mathcal{V}}_1} c_{nd}(x) + \sum_{x \in \mathcal{V}_2 - \hat{\mathcal{V}}_2} c_{ni}(s) \\ & + \sum_{x \in \hat{\mathcal{V}}_1} |\alpha_1(x) - \alpha_2(x)| c_{ns}(x) + \sum_{\substack{\beta_1(e) \cdot \beta_2(e) \neq 0 \\ \beta_1(e) \neq \beta_2(e)}} c_{edc}(e) \\ & + \sum_{e \in \hat{\mathcal{E}}_1} |\beta_1(e) - \beta_2(e)| c_{es}(e) \end{aligned} \quad (4)$$

where $|\alpha_1(x) - \alpha_2(x)|$ and $|\beta_1(e) - \beta_2(e)|$ measure the degree of node substitution and edge substitution, respectively.

According to the likelihood of graph distortions during the graph construction process, we assign different costs to the edit operation of node substitution, edge substitution and edge creation/destruction. The cost function \mathcal{C} is designed as:

$$\mathcal{C} = (c_{nd}, c_{ni}, c_{ns}, c_{es}, c_{edc}) = (\infty, \infty, \lambda_1, 1, \lambda_2) \quad (5)$$

where λ_1 and λ_2 embody the likelihood of different graph distortions. Without prior knowledge, we perform experiments on a training set with various value of λ_1 and λ_2 and select those which maximize the average matching accuracy. Recalling the example in Table 2, the *ECGM* of ground-truth is $f: WIL \rightarrow face1, SPI \rightarrow face2, ANN \rightarrow face3, MAX \rightarrow face4, BEL \rightarrow face5$. Apparently no

node deletion or insertion operation is involved. Also no node substitution operation happens. Therefore, there are five edge substitution operations and two edge insertion operations (edge (3, 5), (5, 3)). The cost of this *ECGM* under our designed cost function \mathcal{C} is: $\gamma_{\mathcal{C}}(f) = 5 + 2\lambda_2$.

A general algorithm to obtain the optimal *ECGM* is based on the A^* method [13]. By applying A^* , we are able to find the best matching by exploring only the most promising avenues, which guarantees a global optimal.

4. SENSITIVITY ANALYSIS

Sensitivity Analysis is common in financial applications, risk analysis, signal processing and any area where models are developed [14]. Good modeling practice requires that the modeler provides an evaluation of the confidence in the model, possibly assessing the uncertainties associated with the modeling process and with the outcome of the model itself. Sensitivity Analysis offers valid tools for characterizing the robustness to noises for a model.

We first introduce two types of simulated noises, coverage noise and intensity noise. Then the impact factor for measuring rank ordinal graph affection is defined to evaluate the sensitivity of our proposed rank ordinal affinity graph to noises. The sensitivity analysis is further discussed in the experiments section.

4.1. Coverage Noise and Intensity Noise

According to the noise investigation, only node substitution, edge substitution and edge destruction/creation are involved in the graph construction process. Thus, we introduce two types of noises, coverage noise and intensity noise, to simulate the real noises.

Coverage noise corresponds to graph edit operation as edges created or destroyed. This type of noise tends to change the topology of the graph. The creation/destruction probability ν_C for each existing or potential edge denotes the coverage noise level. Intensity noise corresponds to changes in the weights of the edges. It has involvement with the quantitative variation of the edges, but with no affection to the topology of the graph. A random value distributed uniformly on the $[-\nu_I, \nu_I]$ range denotes the intensity noise level. As the affinity values are normalized, the intensity change is limited so as to never take the resulting weight outside of the $[0, 1]$ interval. Based on the defined noises, we perform sensitivity analysis in following experiments by investigating the performance of name-face matching with respect to the simulated noises.

4.2. Rank Ordinal Graph Impact Factor

To evaluate the sensitivity of the proposed affinity rank ordinal graph, the impact factor for rank ordinal graph affection is specially defined.

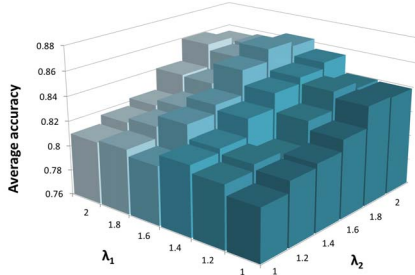


Fig. 1. The average accuracy of name-face matching as λ_1 and λ_2 change. $\lambda_1 = 1.6$ and $\lambda_2 = 1.8$ perform best, with 86.7% average accuracy. This means coverage noise happens not so frequent as intensity noise.

The impact factor function should be consistent with the likelihood of the graph distortions i.e., noises. Therefore, we define the rank ordinal graph impact factor μ in accordance to our definition of the cost function for *ECGM*:

$$\mu = \sum_{x \in \hat{V}_1} \lambda_1 |\alpha_1(x) - \alpha_2(x)| + \sum_{e \in \hat{E}_1} |\beta_1(e) - \beta_2(e)| + \sum_{\substack{\beta_1(e) \cdot \beta_2(e) \neq 0 \\ \beta_1(e) \neq \beta_2(e)}} \lambda_2 \quad (6)$$

λ_1 and λ_2 are the same parameters with Eq.5, and g_1 means the rank ordinal graph before affected by noises and g_2 is the corresponding affected graph.

5. EXPERIMENTS

Experiments are conducted on the same test dataset with [1] including 10 movies of various genres: ‘Mission: Impossible’, ‘Léon’, ‘The Shawshank Redemption’, ‘Revolutionary Road’, ‘Legally Blond’, ‘Devil Wears Prada’, ‘You’ve Got Mail’, ‘Sleepless in Seattle’, ‘Pretty Woman’ and ‘Notting Hill’ (corresponding to the ID of F1 - F10 in Table 3).

5.1. Cost Function of *ECGM*

The costs for different graph edit operations are designed by automatic inference based on a training set of five movies (different from the 10 test movies listed above). Parameters λ_1 and λ_2 in Eq.5 and Eq.6 embody the likelihood of different graph distortions. We perform experiments with various values of λ_1 and λ_2 . The result of average accuracy of name-face matching with respect to λ_1 and λ_2 is shown in Fig.1. $\lambda_1 = 1.6$ and $\lambda_2 = 1.8$ perform best, with 86.7% average accuracy. Beyond or below this, the results deteriorate: the derived parameter values embody the difference of likelihood of node substitution and edge creation/destruction operations from edge substitution. This means that coverage noise (node substitution and edge creation/destruction) are less likely to occur, and the graph topology is relatively stable. We fix $\lambda_1 = 1.6$ and $\lambda_2 = 1.8$ in following experiments.

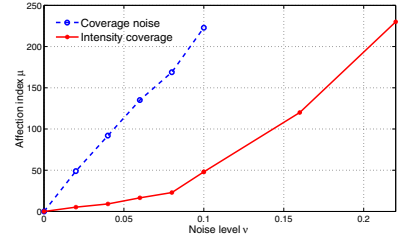


Fig. 2. The rank ordinal graph affliction index μ vs. noise level for coverage noise and intensity noise.

5.2. Evaluation of Name-Face Matching

We compare the name-face matching accuracy of proposed rank ordinal affinity graph + *ECGM* method with the affinity graph + spectral matching method in [1] on the ten test movies. The result is illustrated in Table 3. ‘Our1’ is the method of the proposed rank ordinal affinity matrix with original spectral graph matching, and ‘Our2’ is rank ordinal affinity matrix with *ECGM*. It can be seen that the proposed method improve the performance mostly on the thriller and action movie (F1 and F2), which involve more severe variation of the face pose and illumination. The mechanism of tolerating noises by constructing the affinity rank ordinal graph and employing error correcting concept guarantee the performance of name-face matching. As the name-face matching accuracy is relatively high in [1], the improvement is limited. Therefore, we further show the robustness of our method by introducing simulated noises in sensitivity analysis.

5.3. Sensitivity Analysis

For sensitivity analysis, random coverage and intensity noise of different noise level are generated. We first demonstrate the advantage of constructing affinity graph by rank ordinal matrix by presenting the curve of rank ordinal graph affliction index with respect to coverage and intensity noise in Fig. 2. Fig. 3 illustrates how sensitive the average accuracy of name-face matching is with respect to different coverage noise level and intensity noise level, respectively. Fig. 2 and Fig. 3 show that rank ordinal matrix and the proposed *ECGM* graph matching method basically remain stable within the intensity noise range $\nu_C \leq 0.08$, while the method in [1] tends to fail. However, coverage noise deteriorates the accuracy of name-face matching in our method as well as [1]. We reach the conclusion that the proposed method is very tolerant to random variation to the values of weighted edges, managing to match graphs correctly as long as the topological structure is preserved. This finding is of great importance as according to our observation and experiment results, though the weights of face affinity relations are imprecise, basically the generated name and face affinity graph have the same topology.

Table 3. Performance evaluation of movie summarization

Movie ID	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Total
No. characters	17	14	14	13	20	12	14	21	13	13	151
Zhang											
[1] No. identified	13	10	12	11	16	10	12	17	11	13	125
Accuracy	76.5%	71.4%	85.7%	84.6%	80.0%	83.3%	85.7%	81%	84.6%	100%	82.8%
Our1											
No. identified	13	12	12	11	16	10	12	19	11	13	129
Accuracy	76.5%	85.7%	85.7%	84.6%	80.0%	83.3%	85.7%	90.5%	84.6%	100%	85.4%
Our2											
No. identified	15	12	10	11	18	10	12	19	11	13	131
Accuracy	88.2%	85.7%	71.4%	84.6%	90.0%	83.3%	85.7%	90.5%	84.6%	100%	86.8%

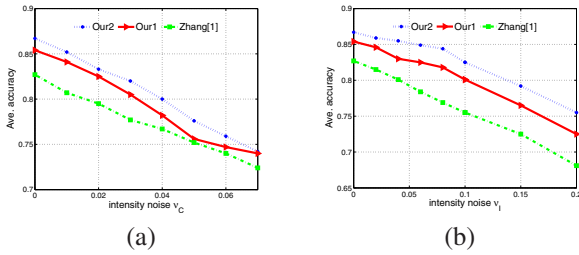


Fig. 3. The average accuracy of name-face matching vs. noise level. (a)for coverage noise (b)for intensity noise.

6. CONCLUSIONS

In this paper, a robust character identification method is proposed and the sensitivity analysis is performed. Experiments demonstrate that the proposed method achieves significant improved performance when the intensity noise is involved. However, the coverage noise which changes the graph topology deteriorates the matching results. In the future, we will be working towards investigating optimal cost function for different movie genres as well as exploiting more character relationships to improve the robustness.

7. ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China (Grant No. 90920303) and the Intel China Research Council sponsored research CRC-2010-22.

8. REFERENCES

- [1] Y.F. Zhang, C.S. Xu, H.Q. Lu, and Y. Huang, “Character identification in feature-length films using global face-name matching,” *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1276–1288, November 2009.
- [2] S. Satoh and T. Kanade, “Name-it: Association of face and name in video,” in *Proceedings of CVPR*, 1997, pp. 368–373.
- [3] J. Yang and A.G. Hauptmann, “Naming every individual news video monologues,” in *ACM Int. Conf. Multimedia*, 2004, pp. 580–587.
- [4] J. Yang and A.G. Hauptmann, “Multiple instance learning for labeling faces in broadcasting news video,” in *ACM Int. Conf. Multimedia*, 2005, pp. 31–40.
- [5] M. Everingham, J. Sivic, and A. Zisserman, “Hello! my name is... buffy automatic naming of characters in tv video,” in *Proceedings of BMVC*, 2006, pp. 889–908.
- [6] M. Everingham, J. Sivic, and A. Zisserman, “Taking the bite out of automated naming of characters in tv video,” in *Journal of Image and Vision Computing*, 2009, pp. 545–559.
- [7] J. Sivic, M. Everingham, and A. Zisserman, “Who are you? - learning person specific classifiers from video,” in *Proceedings of CVPR*, 2009.
- [8] C. Liang, Y.F. Zhang, J. Cheng, C.S. Xu, and H.Q. Lu, “A novel role-based movie scene segmentation method,” in *PCM*, 2009, pp. 917–922.
- [9] J.T. Sang and C.S. Xu, “Character-based movie summarization,” in *ACM MM*, 2010.
- [10] R. E. Hanneman, *Introduction to Social Network Methods*, Online Textbook Supporting Sociology 157., Riverside, CA: University of California, 2000.
- [11] E. Bengoetxea, “Inexact graph matching using estimation of distribution algorithms,” *PhD thesis, Ecole Nationale Supérieure des Telecommunications*, 2002.
- [12] H. Bunke, “On a relation between graph edit distance and maximum common subgraph,” *Pattern Recognition Letters*, vol. 18, pp. 689–694, 1997.
- [13] A. Sanfeliu and K.S. Fu, “A distance measure between attributed relational graphs for pattern recognition,” *IEEE Trans. on SMC*, vol. 13, no. 3, 1983.
- [14] A. Saltelli, M. Ratto, S. Tarantola, and F. Campolongo, “Sensitivity analysis for chemical models,” *Chemical Reviews*, vol. 105, no. 7, pp. 2811–2828, 2005.