Impact of Gaze Analysis on the Design of a Caption Production Software

Claude Chapdelaine, Samuel Foucher and Langis Gagnon,

R&D Department, Computer Research Institute of Montreal (CRIM),
550 Sherbrooke Street West, Suite 100, Montreal (Quebec) H3A 1B9
Claude.Chapdelaine@crim.ca

Abstract. Producing caption for the deaf and hearing impaired is a labor intensive task. We implemented a software tool, named SmartCaption, for assisting the caption production process using automatic visual detection techniques aimed at reducing the production workload. This paper presents the results of an eye-tracking analysis made on facial regions of interest to understand the nature of the task, not only to measure of the quantity of data but also to assess its importance to the end-user; the viewer. We also report on two interaction design approaches that were implemented and tested to cope with the inevitable outcomes of automatic detection such as false recognitions and false alarms. These approaches were compared with a Keystoke-Level Model (KLM) showing that the adopted approach allowed a gain of 43% in efficiency.

Keywords: Caption production, eye-tracking analysis, facial recognition, Keystoke-Level Model (KLM).

1 Introduction

Producing caption for the deaf and hearing impaired required transcribing what is being said and interpreting the sounds being heard. The produced text must then be positioned and synchronized on the image. This is a very labor intensive production task that is expensive and for which turn-around time can be a serious bottleneck. Nowadays, the process can be optimized by using automatic speech recognition (ASR) to reduce the transcribing time. Even so, positioning and synchronizing can remain a demanding task for which, up to now, there is no available solution to assist the captioners.

The goal of this project is to implement and evaluate the feasibility of automatic visual detection techniques (AVDT) to efficiently reduce the time required to position and synchronize text for off-line captioning. However, adding automatic recognition technologies must be carefully implemented to be usable. Indeed, the added ASR technology is effective inasmuch as the error rate is significantly lower than the time needed to transcribe manually. This is also true when adding AVDT. The missed detections, substitutions and false alarms have to be kept to a minimum. Since the actual state-of-the-art technology does not allow us to design software with perfect detection and recognition performances, the potential errors have to be taken into

account at the design phase to enable the user to rapidly correct the situation. This paper presents how the study of human eye-tracking helped us resolved the design challenges associated with AVDT that were implemented and measured with Keystroke-Level Model (KLM). In a first phase of our work [1], the eye tracking study was used to ascertain that the detectable elements obtained from the AVDT would be similar to the relevant region of interest (ROI) perceived by humans. More recently, we revisited the eye-tracking data to further analyze how face detection could be optimized. These results are presented along with the design approaches tested with a proof-of-concept software tool (named SmartCaption) that we implemented for assisting off-line caption production using AVDT.

The paper is organized as follows: Section 2 presents the context of caption production. Section 3 reports on the eye-tracking study done on facial regions of interest. Section 4 explains how face detection was integrated and Section 5 describes the design approaches implemented and how we measured their efficiency. Finally, Section 6 concludes on our work and discusses future plans.

2 Caption

Off-line caption is edited by skilled people (captioners) to establish accuracy, clarity and proper reading rate according to standard guidelines [2]. Besides editing the transcribed text, captioners have to place the text on the image based on their assessment of the value of the visual information. Therefore, this task can be quite labor-intensive; it could require up to 18 hours to off-line captioned one hour of content [2].

The time required to position and to synchronize captions is dependant on the presentation styles which could be either roll-up or pop-up. In roll-up style, captions appear one line at the time in group of two or three lines in which the last line pushes the first line up and out (Fig. 1. left-hand side). They are located in a static region either at the top or bottom of the screen. The roll-up movement indicates the changes in caption line. This style is better suited for programs with high speaking rate and/or with many speakers such as news magazine, sports and entertainment. Positioning only requires switching between upper or lower area of the screen to avoid masking important information.

Fig. 1. Example of pop-up style caption (left) and roll-up style (right).

In pop-up style, caption appears all at once in a group of one to three lines layout (Fig. 1. right-hand side). This style is recommended for dramas, sitcoms, movies, music video, documentaries and children's programs. Each instance of caption must be set out depending of its size which can then be placed anywhere on the image. Typically, the caption is placed not too far from the source of speech (i.e. the speakers) and avoids masking any visual element that may be relevant to the understanding of the content. Furthermore, each caption is given a different place to enable the viewers to perceive the change. These operations demand long execution time and would benefit the most from an efficient automatic positioning and synchronization of the captions.

3 Detecting Visual Content

The expertise of efficiently positioning caption is largely based on human judgment of what is relevant to understand the visual content and avoid masking it with caption. Consequently, to adequately assist captioners in their task, the automatic detections had to identify a significant amount of visual regions of interest (ROI) that would be relevant to humans as well. This assessment was done by us through an eye-tracking analysis involving hearing and hearing impaired people which was based on similar studies [3][4]. For eye-tracking, different algorithms are defined to achieve fixation identification [5]. We used a dispersion-based approach in which fixations correspond to consecutive gaze points that lie in close vicinity for a determined time window. Duration threshold for fixation was set to 250 milliseconds. The dispersion threshold corresponds to a viewing angle smaller or equal to 0.75 degree which was determined by the distance between the centroid of consecutive gaze points. This dispersion threshold is in agreement with the range proposed by Salvucci and Goldberg [6].

The ROI were defined manually and composed of human faces and moving objects. The borders of the frontal view facial ROI were defined based on the performance of an in-house algorithm [7]. The defined ROI were used as targets for fixation measurement (hit). Fixations were computed in terms of percentage of actual hit over potential hit for all ROI of a video. An actual hit (AH) is counted when one participant made at least one fixation in the defined ROI whereas potential hit (PH) is the number of times all the ROI could have been seen by all the participants.

In the first study, all ROI composed a single group that included faces or moving objects. The partial results of the eye-tracking analysis were reported in [1]. ROI were defined in each of the six videos presented to a group of 18 participants (nine hearings and nine hearing-impaired). More recently, a second analysis (using the same eye-tracking data) was conducted which focus on the facial ROI in order to assess its impact on an automatic face detection implementation. We will briefly present the final results for the fixations for all the defined ROI from the first study. Then, we will give the results for the facial ROI.

3.1 Assessment on all the ROI

In our first study, the results suggested that automatic detections could be a viable solution for assessing ROI. Moreover, the final results indicate even more correlation between the automatically detectable items and the human's ROI. As shown in Table 1, the percentage of actual hits for the different types of video is 60.4% and could reach up to 76.8% confirming that AVDT results could potentially be good predictors of regions of interest.

Table 1. Percentage of actual hits on potential hit for ROI.

Video	Type of video	Nb. shots	Length	Nb. of participants	Primary results	Final results
1	Film	43	3m:47s	6	31.9%	45.4%
2	News	37	2m:15s	10	24.3%	46.9%
3	TV magazine	20	2m:14s	6	36.4%	63.8%
4	Documentary	11	2m:38s	8	56.0%	67.7%
5	Film	73	3m:08s	9	53.3%	76.8%
Total		184	13m:17s	39	40.0%	60.4%

We found that faces are crucial for the deaf and hearing impaired, not only for identification purposes but also for lip reading. Petrie *et al.* [8] found that deaf computer users need the whole face and image larger than 120 x 180 pixels to accurately do lip reading. Since faces are so important to these viewers, it implied that an automatic face detection algorithm would be playing an essential role in our implementation. This triggered the need for a more detailed analysis of the facial ROI.

3.2 Assessment of the facial ROI

In the second study, we identified which ROI were facial, but we also indicated which were recurring faces (appearing more than once) and which track (consecutive frames with the ROI) contained a least one frontal view. The study used the fixations found on five of the six videos since one of them did not contain any facial ROI (the sport excerpt was a hockey game).

For all videos, we found high percentages of facial ROI over the total ROI (Table 2), indicating that the number of faces is significantly larger than the number of moving objects. Videos 1, 3 and 4 have more than 90% of the ROI that are facial ones. Many faces can be seen even in these few minute video excerpts (Table 1). This implies many potential faces will be tracked by the face detection algorithm.

This analysis also indicates that in some instances a large percentage of recurring faces is to be expected (Table 2). In videos 1, 4 and 5, which are movie-type more than 70% of the recurring faces are observed, contrary to the television-type videos (2 and 3) with lower percentage. This suggests that recurring faces are frequent enough to influence the interaction design in order to treat them rapidly. It also suggests that the type of production being done could necessitate different detection strategies.

However, further research should be done to investigate if other factors, such as the caption rate and the number of hits on caption, could also characterize production.

Table 2. Percentage of facial ROI.

Video	Nb. ROI	Nb. Facial ROI	% Facial on total ROI	% Recurring faces
1	65	60	92.3%	70.0%
2	52	39	75.0%	28.2%
3	46	43	93.5%	53.5%
4	12	11	91.7%	72.7%
5	76	67	88.2%	76.1%
Total	251	220	87.6%	61,4

However, we need not only to know the amount of data that could be presented to the users but also if this information is relevant to the final viewers. As shown in Fig.2, percentages of hits on potential facial ROI represent a large portion (between 36.5 and 67%) of the overall hits. So, this confirmed that facial ROI are not only numerous, but they are relevant in the production.

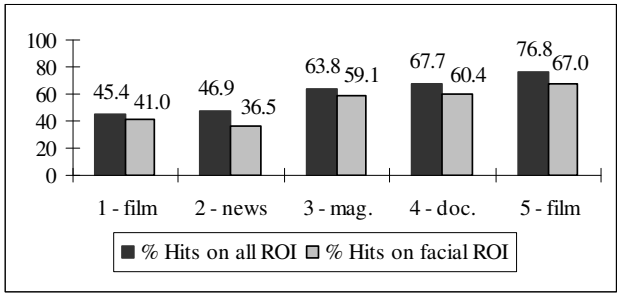

Fig. 2. Percentage of hits on all defined ROI and on facial ROI only.

One of our hypotheses was to focus the design only on the recurring faces. In our primary study, data suggested that certain lower percentages of hits on ROI could be related to recurring faces, since they were less targeted by viewers. But, actual results show there is no correlation between the videos with fewer hits in facial ROI and the ones with more recurring faces. Indeed, higher percentages of recurring faces (video 5, table 2) do not induce more hits on these ROI (video 5, fig. 2.) This implies that all detectable ROI without discrimination have its value and no group of facial ROI should be eliminated from the implementation or interaction. So, strategies need to be adopted to make efficient and usable application.

4 Implementation of the Face Detection Algorithm

The SmartCaption implementation integrates four AVDT: shot, face, text and motion. We will address the interaction design issues for the face detection only. Face detection follows four steps: detection, tracking, normalization and user-assisted face recognition. The near frontal faces are detected using a cascade of weak classifier [9][10]. The tracking is done by a particle filter technique where the particle weight for a given ROI depends on the face classifier response [11]. Face mug shots corresponding to the best frontal views are sampled along the face trajectory and are used for offline recognition. The faces are then normalized by detecting the face center and estimating its width in order to compensate for translation and scaling variations. Face features are derived either from a 2-Dimensional Principal Component Analysis (2DPCA) techniques [12][13][14][15] which directly seeks the optimal projective vectors from face images without preliminary image-to-vector transformation or using SIFT descriptors [16]. The use of SIFT descriptors for face recognition in videos has many advantages: 1) the SIFT signature is robust to various images transformations (rotation, scaling), 2) it is less affected by occlusions and poor face normalization (e.g. presence of background), 3) it has also the potential to reject irrelevant tracks (false detections, etc.). Face recognition is incremental, done on a shot basis and is assisted by the user who is required to validate automatic decisions or reject tracks.

The performance of a detection algorithm is typically measured by comparing the results with a ground truth [19][20], resulting in three possible outcomes: 1) the proper detection is found, 2) the detection is missing or 3) the impostor is wrongly detected (false alarm). The design must enable the user to rapidly acknowledge the good detections and to efficiently recover from error by adding missed items or discarding false alarms.

5 Interaction Design Strategies

Once the detections are done in a batch mode, the captioner must validate the outcomes of the detection algorithms before triggering the production rule engine (reported [21]). As seen in Fig. 3, the captioner is presented with the detected tracks (Fig. 3. Top part with four items) and several actions can be taken; A) accept a new track by giving it a label (id), B) reject a track (false alarm) such as the fourth item (top part), C) correct a track like item 3 that should be changed from “not assigned” to its id which is visible in the accepted track list (bottom part).

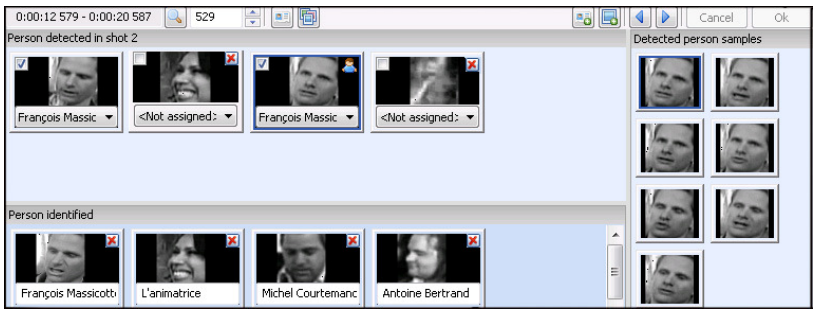

Fig. 3. Face detection interface: Top= track to validate, Bottom=validated track, Right=training set

Some operations require user input (typing or selecting an id) while others require none. For example, track 1 is a detected track with the proper id as is track 4 with the id "not assigned" and it is clearly a false alarm; these results required no user input to accept track 1 and reject track 4. As the captioner navigates from shot to shot, all these operations are used by the algorithm for face recognition. The strategy is to maximize performance on track recognition, reduce false alarm while minimizing user input operations as the task progress thought the video. We tested this explicit training approach on a two-hour film. As seen in Fig. 4, at the beginning there is a peak of recognized and corrected track (together goes up to 60% in the 50 first shots) and false alarm starts at 60%. From 100 shots and more, the performance is stabilizing, and we observed fewer false alarms and corrections are reduced from 20% to 10%.

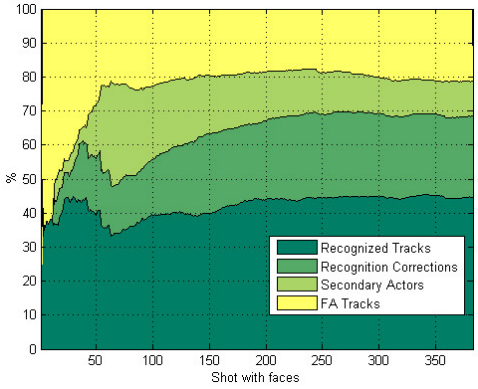

Fig. 4. Percentage by outcomes for face recognition (stack graph)

However, even with an algorithm approach to reduce errors, there is still a great amount of tracks to be validated. For example, in the two-hour film (Fig. 4) 570 tracks were detected over 1,289 shots. The interaction strategies for accepting,

correcting and rejecting tracks have to be efficient i.e. rapidly and easily done. From the facial ROI analysis, we knew that many face tracks would be detected and that all of them have to be treated. The process was done on a shot basis not only to optimize recognition but also to present the user with small chunk of tracks.

We implemented and tested two interaction approaches. Fist, we used an explicit training approach and later, we adopted an implicit training approach. In the explicit training, the user has to select samples (right part in Fig. 3) when creating and correcting track to be trained. In the implicit training approach, the user is only required to respond to one question: "Is this a person that will reappear often?". By answering "yes", the system will automatically add the sample while a "no" answer will keep the person in the track list (as a secondary actor) but the sample will not be used for automatic face recognition.

Obviously, the explicit training approach required more user input and was also more complex since it demanded knowledge on how to choose track for training and what would be the best sample. We measured the gain between the two approaches using a KeyStroke-Level Model (KLM) [22] task analysis that predicts the execution time of a specified interaction implementation based on a sequence of actions.

Table 3. Example of KLM sequences for the two approaches.

Explicit Training Approach			Explicit training Approach		
KSL	Operations	Time	KSL	Operations	Time
p	move to id button	1.10	p	move to id button	1.10
b	id button	0.10	b	id button	0.10
h	hands on keyboard	0.40	h	hands on keyboard	0.40
m	think (recall) name	1.20	m	think (recall) name	1.20
t	type name	5.04	t	type name	5.04
p	move to ok	1.10	p	move to ok	1.10
b	ok button	0.10	b	ok button	0.10
p	move on pull-down	1.10	p	question : is recurrent	1.10
b	on pull-down button	0.10	m	think answer	5.04
p	move in name list	1.10	p	move to y/n button	1.10
m	choose name	1.2	b	y/n button	0.10
b	release mouse	0.10			
p	move on face icon	1.10			
b	on face icon	0.10			
p	move in training icons	1.10			
m	choose good sample	1.20			
p	move to good sample	1.10			
b	on good sample	0.10			

As stated by Kieras [22], seven actions are monitored with their proposed experimental time measurements. There are: p (moving the mouse to a target) 1.1 second, b (click/release) mouse .1 second, h (hands on keyboard) .4 second, t (typing time TIMES estimate number of characters) n*.28 second, m (mental time to decide/perceive) 1.1 second. Table 3 gives an example of the sequence of KLM used to measure the addition of a new id for both approaches.

Table 4. KLMoutput for the two approaches to validate key face (in seconds).

Approach	Go to shot	Name track	Correct track	Reject Track	Total
1	220.8	641.6	485.5	9.6	1357.5
2	64.8	597.2	100.8	9.6	772.4
Gain	156.0	44.4	384.7	0.0	585.1

From the five video with 125 detected tracks on 54 shots, we obtained a time gain of 43%, by reducing the total amount of time required to validate from 1,357.5 seconds to 772.4 (Table 4). Explicit training required a sample selection for adding and sometimes for correcting. The implicit approach only required the operations limited to answering the question. Time was also reduced by improving navigation with buttons that enable the user to go directly to shots with detected track. This facilitated the validation of the detected tracks while identifying shots where faces would potentially be added (to process missed detections).

6 Conclusion

The second eye-analysis study presented here helped us identifying not only the amount of the tracks that would be treated by a face detection algorithm, but it also taught us that they are relevant to the deaf and hearing impaired. Thus, all tracks should be kept in the production. The study also suggests that this amount of faces and their importance can vary depending on the style of content being produced. It seems that the nature of facial data for films and documentaries could be different than those for television programming. Usability and detection performances can benefit from the knowledge gained from eye-tracking analysis and the KLM measurement. The goal of our future work is to further optimize the usability for a real world captioning application. We plan to do more testing on a wide variety of content of various lengths and to use KLM to identify potential groups of interaction operations that can be redesigned to reduce production time.

Acknowledgment

The authors would also like to thank their colleagues in the Vision and Imagery team: Mario Beaulieu, Valerie Gouaillier, David Byrns and Marc Lalonde for their implication in the previous phase of this project.

References

1. Chapdelaine. C., Gouaillier. V., Beaulieu M., Gagnon L.: Improving Video Captioning for Deaf and Hearing-impaired People Based on Eye Movement and Attention Overload, Proc. of SPIE, Volume 6492, Human Vision & Electronic Imaging XII (2007)
2. CAB: Closed Captioning Standards and Protocol, CAB eds., 66 p. (2004)
3. D'Ydewalle, G., Gielen, I.: Attention allocation with overlapping sound, image and text, In Eyes movements and visual cognition, Springer-Verlag, pp 415-427 (1992)
4. Jensema, C., Sharkawy, S., Danturthi, R. S.: Eye-movement patterns of captioned-television viewers, American Annals of the Deaf, 145(3), pp. 275-285 (2000)
5. Josephson, S.: A Summary of Eye-movement Methodologies, http://www.factone.com/article_2.html (2004)
6. Salvucci, D. D., Golberg, J. H.: Identifying fixations and saccades in eye-tracking protocols, In Proc. of ETRA, New York, pp. 71-78 (2000)
7. Foucher, S., Gagnon, L.: Automatic Detection and Clustering of Actor Faces based on Spectral Clustering Techniques, 4th Canadian Conference on CRV, Montreal (2006)
8. Petrie, H., Fisher, W., Weimann, K., Weber, G.: Augmenting Icons for Deaf Computer Users, Proceedings of the 2004 Conference on Human Factors in Computing Systems. E. Dykstra-Erickson and M. Tscheligi. Editors, ACM, New York, pp. 1131–1134 (2004)
9. Viola, P., Jones, M.J.: Rapid object detection using a boosted cascade of simple features, IEEE CVPR, pp, 511-518 (2001)
10. Lienhart, E., Maydt, J.: An extended Set of Haar-like Features for Rapid Object Detection, in IEEE ICME (2002)
11. Verma, R.C., Schmid, C., Mikolajczyk, K.: Face Detection and Tracking in a Video by Propagating Detection Probabilities, IEEE Trans, On PAMI, Vol. 25, No. 10 (2003)
12. Yang, J., Zhang, D., Frangi, A.F., Yanf, J.: Two- Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition, PAMI, 26(1): 131-137 (2004)
13. Kong, H., Li, X., Wang, L., Teoh, E.K., Wang, J.-G., Venkateswarlu, R.: Generalized 2D Principal Component Analysis, IJCNN, Montreal, Canada (2005)
14. Zhang, D., Zhou, Z.-H., Chen, S.: Diagonal principal component analysis for face recognition, Pattern Recognition, 39(1):140-142 (2006)
15. Zuo, W.-M., Wang, K.-Q., Zhang, D.: Assembled Matrix Distance Metric for 2DPCA-Based Face and Palmprint Recognition, In Proc. Of ICMLS, pp. 4870-4875 (2005)
16. Lowe, D. G.: Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, pp. 91-110 (2004)
17. Chen, X., Yuille, A. L.: Detecting and reading text in natural scenes, Proc. CVPR 2004, Vol. II, pp. 366-373 (2004)
18. Lucas, B., Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision, Proc. of 7th Int. Joint Conference on AI, pp. 674-679 (1981)
19. Sherrah, J.: False alarm rate: a critical performance measure for face recognition, Automatic Face and Gesture Recognition, Proceedings Sixth IEEE International Conference on Volume Issue, pp 189 – 194 (2004)
20. Phillips, P.J., Scruggs, W.T., O'Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 Large-Scale Results, NISTIR 7408 (2007)
21. Chapdelaine, C., Gouaillier, V., Beaulieu, M., Gagnon, L.: Designing Caption production rules based on face, text and motion detections, Volume 6806, Human Vision & Electronic Imaging XIII (2008)
22. Kieras, D.E.: Using the Keystroke-Level Model to Estimate Execution Times, The University of Michigan, http://www.pitt.edu/~cmlewis/KSM.pdf (1993)