Electronic Slide Matching and Enhancement of a Lecture video

G. Gigonzac, F. Pitié, A. Kokaram[1]

[1] Sigmedia Trinity College Dublin, Ireland,
guillaue@tcd.ie, fpitie@mee.tcd.ie, anil.kokaram@tcd.ie

Keywords: e-learning, slide detection, tracking

Abstract

This paper presents an automatic method to enhance video presentations for distance learning applications. From a material recorded by a fixed, non professional camera, the system matches the slides displayed during the presentation with their electronic versions. The process to achieve slide recognition consists of two phases. In the first phase, the location where the slides are displayed is located by colour matching. Then a shot detection is performed in the display area and a frame is selected for each slide displayed in the video. The second phase consists of matching the frames previously selected to the electronic version of the slides. Using correlation measure, a likelihood is computed for each electronic slide to correspond to the slides displayed in the frames selected. A prior distribution is then defined to model the probability of each possible slide transition. Finally the most probable sequence of slides displayed in the video is determined using the Viterbi algorithm. The results show that the method presented is robust against luminance conditions, occlusion by the lecturer and can be performed for a large variety of presentations.

1 Introduction

With the development of telecommunications and multimedia technologies, distance learning systems are becoming more and more popular [1]. The aim of these systems is to record presentation talks and to broadcast their content over the Internet to distant users. The problem with directly broadcasting the recorded material is that the video is likely to be very dark because the environment suitable for local viewing is not amenable to a good video recording. Light levels can be too low for instance, and the camera is unable to be placed close to the presenter. In these conditions it is often difficult to follow the broadcasted talk. Of course special purpose broadcasting rooms for e-learning content are sometimes exploited in large institutions e.g. universities, but these rooms are not available to all potential users. The idea in this paper is to create editing technologies that reduce the need for manual editing of this kind of video and automatically enhance the key content features.

Previous work has considered this problem in part and typical pre-processing involves segmenting the lecturer from the background [4, 8, 12], localising the projection area [5, 7, 10], finding the slide transitions [4] and matching the observed slides in the videos to their electronic versions [4–7, 10]. Localising the projection area is key to a good automatic parsing of the presentation and this paper presents a technique that resolves many of the robustness issues of previous work [5, 7, 10]. Identifying the slide that is in the field of view is clearly of importance since i) it gives some degree of interactivity because the user can navigate in the video and jump to the desired slide and ii) it provides access to a high quality picture of the slide which offers a similar experience as in presentation room. The work presented here exploits a robust slide change detection process to achieve slide identification. In addition, this paper explores the possibility of exploiting the slide i.d. knowledge to actually change the original footage itself. Given the knowledge of the slide i.d, a high resolution version of the electronic slide could be inserted into the field of view to replace the low-resolution projection of the slides. An example of such a composition is illustrated on Fig. 1. This is a significant advance on previous ideas and combines concepts from post-production with automatic video parsing.

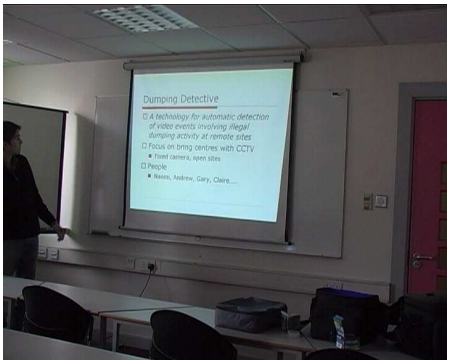
The second step consists in matching the slides displayed in the video with the high definition electronic slides used during the presentation. That for, the presentation is modelled by a HMM of order 1 similarly to [6]. The likelihood of the states are computed using a simple correlation measure. The transitions probabilities are set according to the natural sequential order of the presentation. The most probable path, that is to say the most probable sequence of slides displayed during the presentation, is determined using the Viterbi [11] algorithm. In order to cope with possible occlusions by the lecturer, it is also proposed to enhance the HMM with an extra set of ‘occlusion’ states.

Organisation of the paper.

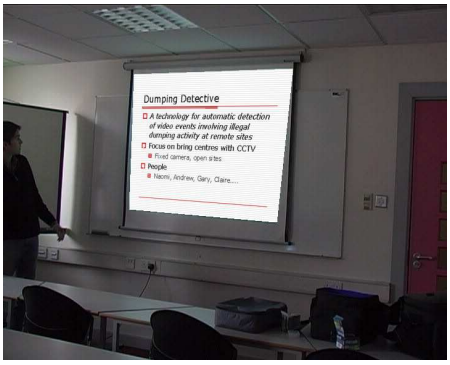
The first part of the paper presents the technique used to detect the slide transitions and the display area. The second part describes the HMM model and likelihood used to compute the most probable path. The paper ends with results on various footage of actual lectures.

2 Slide Transition and Display Area Detection

Techniques to find the display area in the video usually employ colour segmentation. The idea is to extract parts in the frames which have the same colour distribution than the original electronic slides. The reasoning behind is that presentations are usually written on a fixed background image, or *template*,



(a)



(b)

Figure 1: *(a) Original image. (b) Composition with electronic slide.*

which presents constant colours characteristics throughout the video. Looking for these colours in the frames give then an indication of the display area. In practice however, the displayed colours never match the original colours. A more sensible approach is therefore to use the actual displayed colours in the video. To have access to these colours, a first guess of the display area (denoted as A_1), is found by detecting where slide transition appear in the video. This is done using a coarse temporal filter as described hereafter. From this first guess A_1 it is possible to extract the colour distribution of the template. Then a colour segmentation based on these observed colours is then employed to refine the delineation.

2.1 Rough Slide Changes Detection

The first step of the process requires then to find when and where slide changes happen in the video. Slide transitions could be treated as shot transitions and a commonly used technique to detect shot cuts is to use colour histograms [13]. However, for this application, the colour histograms may not be very different from one slide to the other. The changes are usually in fact very small since only the text on the slides change. It is then more interesting to measure directly frame to frame differences. To avoid a full treatment of the video,

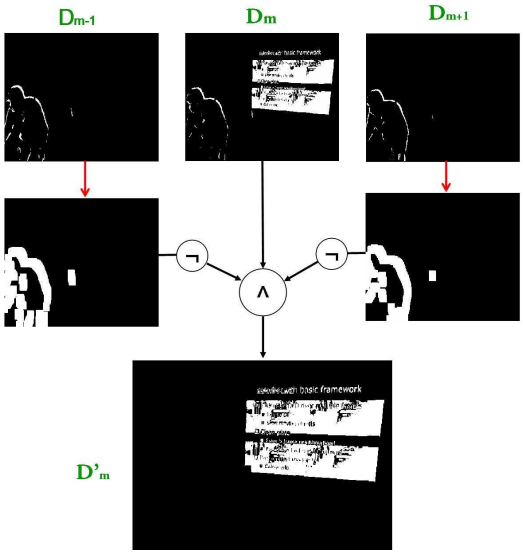


Figure 2: *Removal of the changes due to the lecturer*

the differences are computed for frames separated by 300ms. In the following, consecutive frames I_m and I_{m+1} are thus in fact separated by 300ms. The frame difference is realised in the YUV space as follows:

$$D_m(x,y) = \begin{cases} 1 \text{ if } |Y_m(x,y) - Y_{m+1}(x,y)| + \\ \quad |U_m(x,y) - U_{m+1}(x,y)| + \\ \quad |V_m(x,y) - V_{m+1}(x,y)| > T \\ 0 \text{ otherwise} \end{cases} \quad (1)$$

where T is a thresholded fixed to 40 out of 256 levels. The frame changes have three major causes: 1) movements of the lecturer, 2) noise due to the camera or luminosity variations and 3) the slide transitions.

The movements of the lecturer manifest as continuous changes in the video. The continuity of the changes can help to predict where they occur in D_m by observing where they also occur in D_{m-1} and D_{m+1}. To account for the displacement of the lecturer, frame differences D_{m-1} and D_{m+1} are spatially expanded by morphological dilatation. The dilated maps form a mask for the lecturer motion as it is illustrated in Fig. 2. Then subtracting this mask to D_m yields a frame difference map D_m which does not contain the lecturer motion.

The resulting map now contains principally noise due to luminosity changes and differences due to slide transitions. Thus a simple thresholding is enough to detect if the frame corresponds to a slide transition:

$$\sum_{x,y} D_m(x,y) < T_D \quad (2)$$

The activity threshold T_D has been set to 100 pixels for PAL images. The frame differences that are flagged as significant

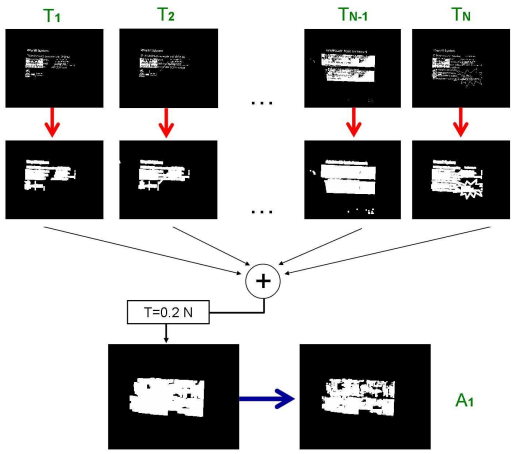


Figure 3: *Computation of the impulsive change map: the first row shows the change maps $T_1, ..., T_N$ corresponding to slide transitions. The second row is the dilation of the maps of the first row (a dilation is represented by a red vertical arrow). The dilated images are then summed and thresholded to produce the map on bottom left. This map is finally eroded to form the map* A_1 *(the erosion is represented by a blue horizontal arrow).*

slide transitions are denoted as $T_1, ..., T_N$. In order to remove the noise, the locations of changes in $T_1, ..., T_N$ are averaged as illustrated in Fig. 3. First the transitions are dilated with a structuring element $E = 10 \times 10$ pixel square. The dilated transitions are then summed and averaged to pick only the location where the changes occur more than 20% of the time. The map A_1 obtained is finally eroded with the structuring element E to form the first approximation of the display area.

2.2 Template Segmentation

With this first delineation of the display area, it is possible to obtain the colour distribution of the displayed colours. An extra processing step is however required: the display area contains a mixture of the template and extra text and pictures that are not part of the actual template colours and should be therefore removed before establishing the final colour distribution. Since the template remains identical during the slide transitions (in contrast to the text and images) the changes that appear in T_n and T_{n+1} are removed from the map A_1 as illustrated in Fig. 4.

For each slide, a colour segmentation the template can now be done. To account for possible occlusion problems, the last estimate A_2 of the display area is defined as the area recovered by the template colour segmentation for more than 20% of the slides. This means that the colour segmentation is performed for each frame and that a pixel belongs to the display area if it has been segmented as template pixel for more than 20% of the frames. Finally the four edges of the display area can then be

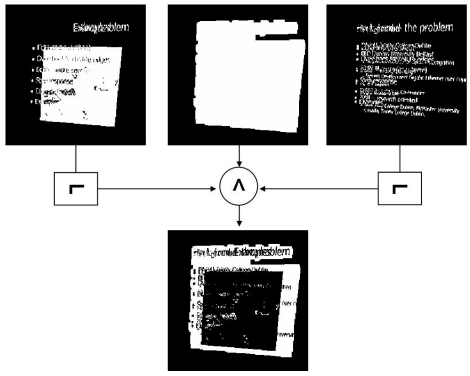


Figure 4: *Background Definition. In the first row the side figures are the change maps of two consecutive transitions, the figure in the middle is* A_1*. The map* A_1^n *on the bottom is obtained by removing the changes T_n and T_{n+1} to the map* A_1*.*

recovered using the Hough transform of A_2. The corners of the display area are defined as the intersections of those four lines.

2.3 Final Slide Transition Detection

In order to identify slides in the following, accurate slide transitions are needed. The previously proposed change detection mechanism is not robust enough for the kind of accuracy which is required here. To enhance the rough transition detection, the slide transition detection is computed based on the knowledge of the display area and using a larger temporal window. The detection is performed in a similar way to shot detection techniques [2]. To detect if a transition occurs between frame I_n and I_{n+1} one considers the neighbouring images $I_{n-4}, ..., I_n$ and $I_{n+1}, ..., I_{n+5}$ (where frames are distant by $300ms$). Transitions are found by observing the magnitude of frame variations $H(n)$ in the display area throughout the video:

$$H(n) = \sum_{(x,y)\in A_2} \left| I_n^-(x,y) - I_n^+(x,y) \right| \qquad (3)$$

As illustrated in Fig. 5, peak values of $H(n)$ indicate a significant slide transition. Noise due to minor changes is first removed by only considering the scores $H(n) > 0.05 \times H_{max}$, where H_{max} is the score obtained for I_n^- fully black and I_n^+ fully white.

3 Slide Identification

The slides displayed in the video between two slide transitions (see 2.3) are now required to be matched with their electronic version. Since the video is recorded from a presentation it is likely that the slides are displayed in an increasing order. The two parameters retained to match the slides displayed in the video are thus their similarity with the electronic slides but also the order in which they are displayed. In order to implement those dependencies, an Hidden Markov Model

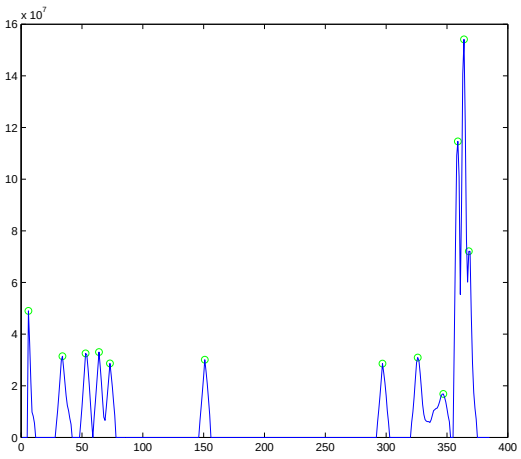


Figure 5: *Graph $H(n)$. The peaks recovered as slide transitions are circled in green.*

(HMM) has been used to model the presentation [9]. The likelihood probabilities of the states are defined from the similarity measure, the transitions from the probability for the presenter to go from one slide to another. Finally the most probable sequence of slides displayed is estimated using the Viterbi algorithm.

3.1 The HMM Model

Denote as O_n the set of frames observed between the slide transitions T_{n-1} and T_n. Sometimes two slides transitions might occur within less than two second. These kind of shots contain very little information than can be accurately used and are therefore discarded. The remaining observations are then denoted as $O_1, ..., O_R$ in the following.

For a presentation containing M electronic slides, $2M$ states are defined in the model. The states $S_1, ..., S_M$ correspond to the M electronic slides. The states $S_{M+1}, ..., S_{2M}$ all correspond to virtual states of occlusion. A virtual state corresponds to the case when no recognised slide is currently displayed. As opposed to the HMM model used in [6], where only one occlusion state is considered, several states have been introduced here. The reason behind is that with only one occlusion state, it is impossible to know the slide number prior to occlusion. For example, the prior probability for the sequence of states 1-Occlusion-2 is the same as for the unlikely sequence 1-occlusion-1000. The idea introduced here, is to encapsulate the slide number information in the occlusion state. Each occlusion state S_{M+i} corresponds to the situation where an unknown slide is displayed, but where the last known slide displayed is i. This enables the model to be a HHM of order 1 (each state only depends on the previous state) while keeping the information of the last slide displayed even in the case of an unknown slide.

Denote as $(q_1, \cdots, q_R), x_i \in [1, 2M]$ the states displayed during the observations $(O_1, \cdots, O_R)$ (i.e. $q_r = S_m$ if the slide S_m is displayed during observation O_r). Finding the best sequence of the slides displayed in the video consists in finding the MAP among the possible sequences, i.e.

$$\arg\max p(q_1, \cdots, q_R | O_1, \cdots, O_R) \tag{4}$$

The presentation the model is assumed to a Markov chain of order one:

$$p(q_{r+1} | q_1 \cdots, q_r) = p(q_{r+1} | q_r) \tag{5}$$

The MAP can thus be found using the Viterbi algorithm knowing the likelihood $L(O_r|S_m) = p(O_r|q_r = S_m)$ and the transition priors $T_{i,j} = p(q_{r+1} = S_j | q_r = S_i)$ (independent of r in the model). For computational reasons, the quantities considered in the following are the negative log likelihood $E(O_r|S_m) = -\log(L(O_r, S_m))$ and the negative log prior $V_{i,j} = -\log(T_{i,j})$.

3.2 Transitions Prior

For two states S_i and S_j the transition $V_{i,j}$ is defined as follow.

- if $i \leq M$ and $j \leq M$, the two states correspond to electronic slides:
$$\begin{cases} \text{if } j = i+1, V_{i,j} = 0 \\ \text{if } j = i, V_{i,j} = 0.05 \\ \text{if } j = i-1, V_{i,j} = 0.05 \\ \text{if } |i-j| = 2, V_{i,j} = 0.1 \\ \text{if } |i-j| > 2\,, V_{i,j} = 0.15 \end{cases}$$
- if $i \leq M$ and $j > M$, $V_{i,j}$ is the probability to go from an electronic slide to an unknown slide:
$$\begin{cases} \text{if } j = i+M, V_{i,j} = 0.1 \\ \text{if } j \neq i+M, V_{i,j} = +\infty \end{cases}$$
- if $i > M$, the slide of origin is unknown,$V_{i,j}$ is defined from the last electronic slide displayed : $V_{i,j} = V_{(i-M)j}$

The trellis representing the HMM is illustrated in Fig. 6.

3.3 Likelihood

In order to compute the log likelihood $E(O_r|S_m)$ five equidistant frames $F_r^i, i \in [1, 5]$ are selected from observation O_r and compared to S_m using a correlation score. In order to use the correlation measure, the display area of the frames F_r^i and the electronic slides S_m are first processed to be of the same shape and size.

The deformation between the electronic slides and the display area is called a projective transformation. This transformation can be modelled by a matrix of eight unknown parameters [3]. These eight parameters are estimated by matching the corners of the display area with the corners of the electronic slides (each corner gives two equations). Since the correlation is based on pixel matching, it is preferable for the two images compared to have a similar resolution. Since the projective transformation decreases the resolution, it has been chosen

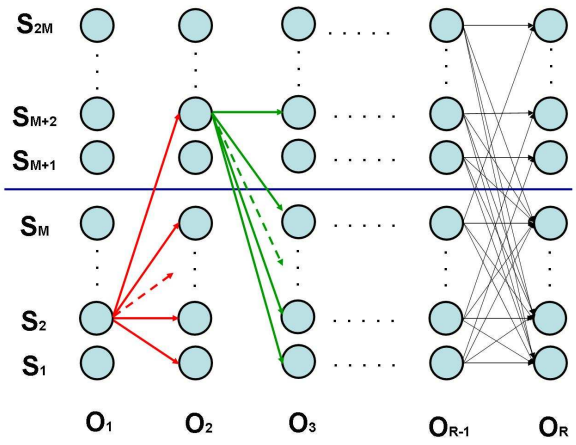


Figure 6: *Model of the trellis used to represent the HMM. The sates $S_1, ..., S_M$ correspond to electronic slides. The sates $S_{M+1}, ..., S_{2M}$ correspond to an unknown electronic slides displayed knowing the previous slide displayed is respectively $S_1, ..., S_M$. The red arrows illustrate the possible (i.e. with a non infinite cost) transitions to go from a state $q_r = S_m, m \leq M$ to the state q_{r+1}. The green arrows illustrate the possible transitions to go from a state $q_r = S_m, m > M$ to the state q_{r+1}.*

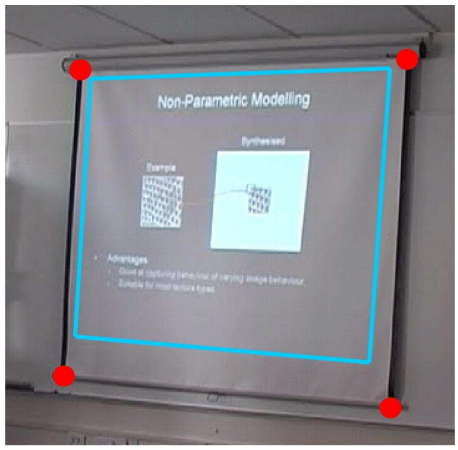


Figure 7: *The red dots show the corners found automatically. The blue lines form the contour of the actual display area not visible to the eye*

to warp the electronic slides (of higher resolution) onto the display area. The warped slides obtained are denoted $S'_1 \cdots S'_M$.

A correlation score is computed between each frame $F_r^i, i \in [1, 5]$ of observation O_r and each electronic slide S'_m. This measure is defined as,

$$C(F_r^i, S'_m) = \frac{\sum_{(x,y)\in A}[F_r^i(x,y) - \bar{F_r^i}][S'_m(x,y) - \bar{S'_m}]}{\sqrt{\sum_{(x,y)\in A}[F_r^i(x,y) - \bar{F_r^i}]^2 \sum_{(x,y)\in A}[S'_m(x,y) - \bar{S'_m}]^2}} \quad (6)$$

where A denotes the pixels corresponding to the display area and $\bar{F_r^i}$ (resp. $\bar{S'_m}$) is the average value of F_r^i (resp. $S'm$) in the display area. The correlation score of the observation O_r is defined as the maximum score obtained for $F_r^i, i \in [1, 5]$,

$$C_r(m) = \max_{i\in[1,5]} C(F_r^i, S'_m) \quad (7)$$

For $m > M$ the correlation score $C_r(m)$ is set to the average correlation score of 0.

From the correlation score, the log likelihood for an observation O_r to be in state S_m is given by,

$$E(O_r|S_m) = 1 - C_r(m) \quad (8)$$

In order to take into account the fact that the presentation is likely to begin by the first slide of the presentation, the likelihood are enhance for the slide S_1 and S_{M+1}: $E(O_1|S_1) = C_1(1) - 1.1$ and $E(O_1|S_{M+1}) = 0.1$.

Finally the calculation of the optimal state sequence is performed by the Viterbi algorithm.

4 Results

Tests have been performed for eight recorded presentations. The video used were 576×720 RGB frames at a rate of 10 images per second. Seven different power point presentations were used with different templates and different animations (such as fade, solve and progressive appearance). The videos were recorded with different angle of camera and different lightening conditions. The first part presents the results for the display area detection and the slide transition detection. The second part shows the final results of the slide matching.

4.1 Display Area and Observations

The display area was correctly recovered for seven out of eight presentations. In the eighth presentation, the area recovered corresponds to the entire screen instead of the location where the slides are displayed Fig. 7. This comes from the fact that the template used in this presentation is totally black and the colour of the template is very difficult to differentiate from the screen even for human eyes.

The next results, in Table 1, shows the accuracy of the observation chosen. These observations directly depend on the slide transition detection performed in 2.3. In order to measure the accuracy of the system the observations were classified in five types: i) an observation is classified as *correct* if it corresponds to a part of the movie where a unique slide is displayed and if this slide is not displayed in the previous observation, ii) an observation is classified as *repeated* if it corresponds to a part of the movie where a

presentation	1	2	3	4	5	6	7	8
# correct	6	16	5	8	8	12	25	28
# repeated	2	3	2	1	4	5	7	6
# false	0	0	0	0	1	1	0	0
# missed	0	0	1	0	0	0	1	0

Table 1: Slide transition detection results on 8 presentations. Notation: #correct denotes the number of electronic slides found as correct (see text).

presentation	1	2	3	4	5	6	7	8
# matched	8	17	6	7	11	8	29	19
# mismatched	0	2	1	2	2	10	3	15
# slides	26	25	18	28	26	18	20	31
rate (%)	100	89	86	78	85	44	91	56

Table 2: Slide matching results on 8 presentations.

unique slide is displayed and if this slide is displayed in the previous observation, iii) an observation is counted as *false* if it corresponds to a part of the movie where several slides are displayed, iv) if a slide is displayed in the video during more than two second with no observation corresponding to it, an observation is considered *missing*.

4.2 Path Matching

The number of correct matches between an observation and an electronic slide are evaluated in Table 2. The correct and repeated observations are both counted as matched if the electronic slide is correctly recovered, wrong otherwise. The false observations are automatically counted as mismatched (even if one of the slide that the observation contains is recovered).

The number of mismatched observations in presentation 6 is due to the occlusion with the lecturer. In four consecutive observations, the correlation measure is biased because the lecturer is in front of the display area. As there are few observations in the video, this error spreads to the other observations because of the prior on transitions. The presentation 8 is the presentation for witch the display area is mismatched. Thus the correlation performed to compute the likelihood is no more reliable. However one can see that more than half of the slides are still correctly matched.

5 Final Comments

The detection of the display area is a key point in the process of lecture video enhancement. The results obtained show that the method presented is efficient in difficult lightening conditions and for a large variety of presentations. In general some semi-automatic user assisted tool for slide localisation is required to recover from pathological cases such as presentation 8. Such an algorithm is outside the scope of this paper, but a result is shown in Fig. 8 to illustrate our current work in that direction. The slide matching described in this paper allows to find the slides displayed in the presentation and replace them by their electronic version. Future work include tracking the lecturer to find if occlusion has happened and compensate to display the electronic slides.

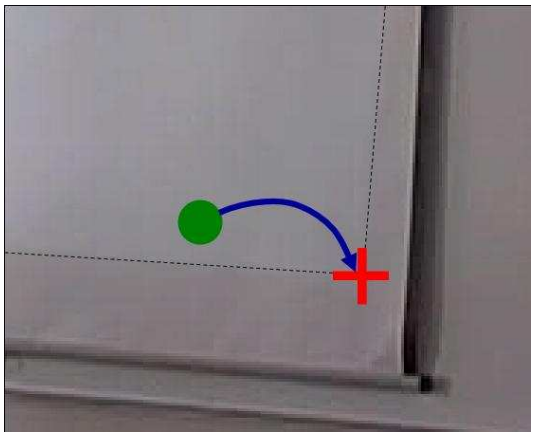


Figure 8: *The green dot corresponds to the point selected by the user. The corner of the projection area, represented by a red cross, is automatically recovered.*

Acknowledgements

The work described in this paper was supported by the Science Foundation of Ireland and Sigmedia, Trinity College, Dublin.

References

[1] G. Abowd, C. G. Atkeson, A. Feinstein, C. Hmelo, R. Kooper, S. Long, N. Sawhney, and M. Tani. Teaching and learning as multimedia authoring: the classroom 2000 project. *IBM Systems Journal, Special issue on Pervasive Computing*, 38(4):508–530, 1999.

[2] H. Denman. *Discontinuity Analysis for Video Processing*. PhD thesis, University of Dublin, Trinity College, October 2006.

[3] F. Dibos. Projective analysis of 2-d images. *Image Processing, IEEE Transactions*, 7:274 – 279, 1999.

[4] B. Erol, J. Hull, and D. Lee. Linking multimedia presentations with their symbolic source documents: Algorithm and applications. *International Multimedia Conference, Proceedings of the eleventh ACM international conference on Multimedia*, pages 498–507, 2003.

[5] Q. Fan, K. Barnard, A. Amir, A. Efrat, and M. Lin. Matching slides to presentation videos using sift and scene background matching. *International Multimedia Conference,Proceedings of the 8th ACM international*

workshop on Multimedia information retrieval, pages 239–248, 2006.

[6] Q. Fan, K. Barnard, A. Amir, R. Swaminathan, and A. Efrat. Temporal modeling of slide change in presentation videos. *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 1:989–992, 2007.

[7] T. Liu, R. Hjelsvold, and J. R. Kender. Analysis and enhancement of video of electronic slide presentations. *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, 1:70–80, 2002.

[8] C. Ngo, T. Pong, and T. Huang. Detection of slide transition for topic indexing. *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, 2:533–536, 2002.

[9] N. Rea. *High-level Event Detection in Broadcast Sports Video*. PhD thesis, University of Dublin, Trinity College, October 2004.

[10] T. F. Syeda-Mahmood. Indexing for topics in video using foils. *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, 2:312–319, 2000.

[11] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260267, 1967.

[12] F. Wang, C. Ngo, and T. Pong. Gesture tracking and recognition for lecture video editing. *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 3:934–937, 2004.

[13] W. Zhao, J. Wang, D. Bhat, K. Sakiewicz, N. Nandhakumar, and W. Chang. Improving color based video shot detection. *Multimedia Computing and Systems, 1999. IEEE International Conference*, 2:752–756, 1999.