Robust Alignment of Presentation Videos with Slides

Presented by
Yekaterina Kharitonova

October 8, 2010

* Note: The images and concepts in this presentation were taken taken from the original paper "*Robust Alignment of Presentation Videos with Slides*" by Xiangyu Wang and Mohan Kankanhalli

Motivation for slide matching

- Index videos by slides for searching and browsing
- Help in understanding the lecture by showing the corresponding slide
- Improve the quality of the video by projecting the slide back into the video

Video styles

(a) Style 1: switch between slide and presenter

(b) Style 2: both slide and presenter

(c) Style 3: slide in background

Video-slide alignment overview

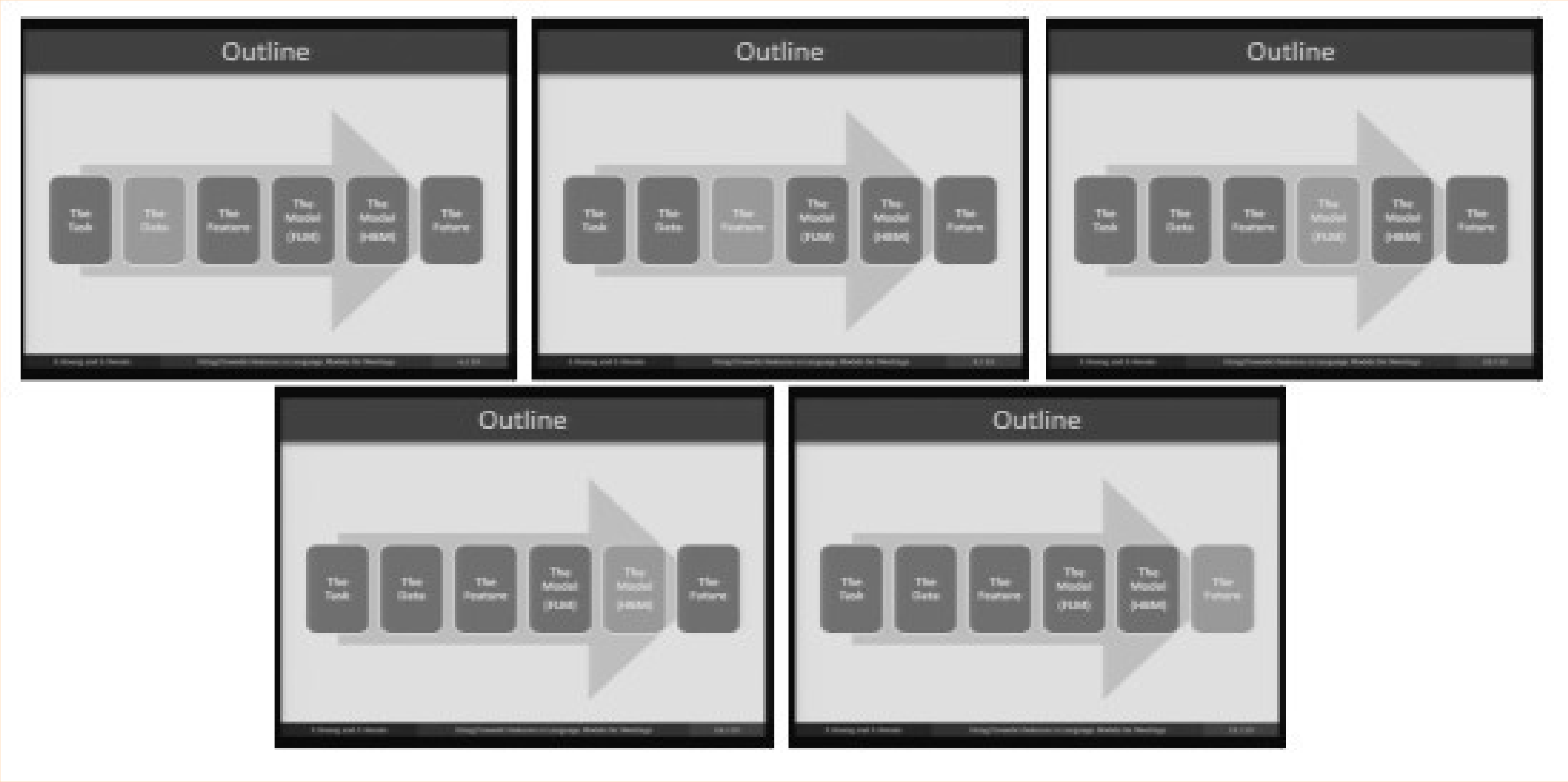
Combine both the *SIFT keypoint features* and *color features*, and use the *texture features* as complement to improve the slide-to-video alignment that can work for different video styles.

Descriptors: SIFT keypoints' advantages

- SIFT keypoints perform reliable matching between different views of a slide
  - across a range of affine distortions
  - change in 3D viewpoint
  - addition of noise
  - change in illumination
- They are invariant to image scale and rotation

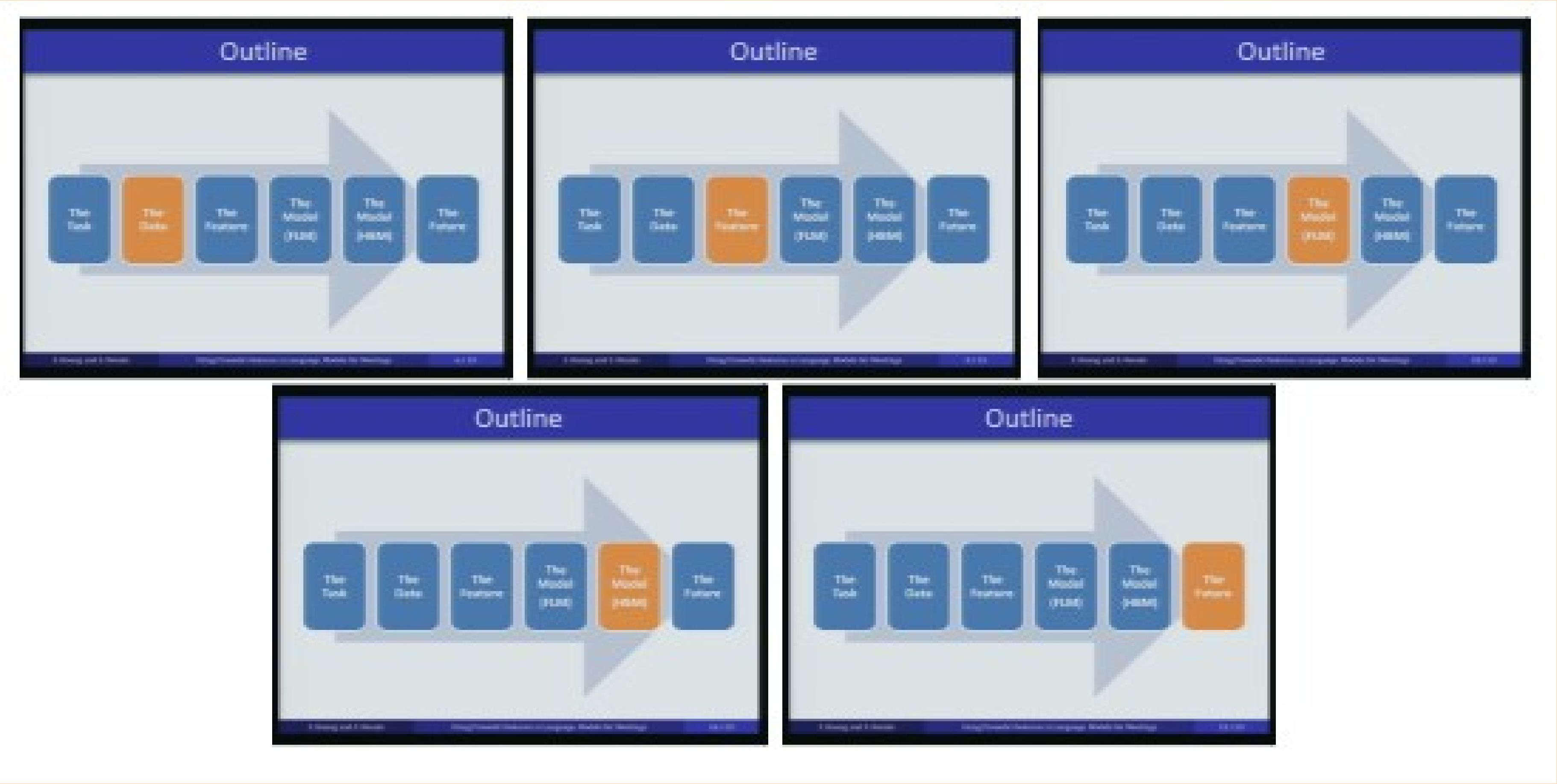
Descriptors: SIFT keypoints' disadvantages (1)

Cannot differentiate images with the identical content but different highlighted sections



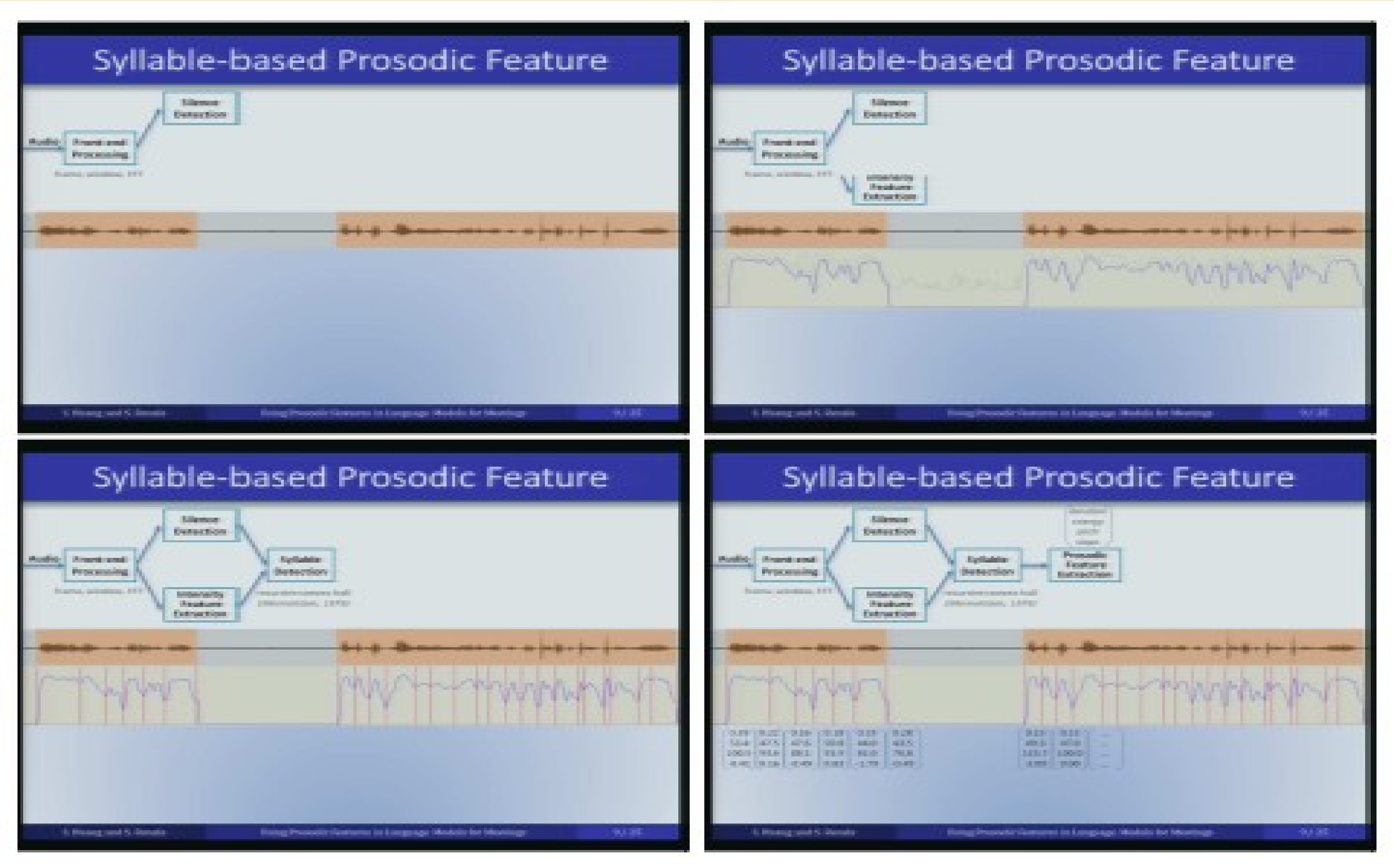
Descriptors: SIFT keypoints' disadvantages (1)

Cannot differentiate images with the identical content but different highlighted sections



Descriptors: SIFT keypoints' disadvantages (2)

- Cannot handle animated slides well

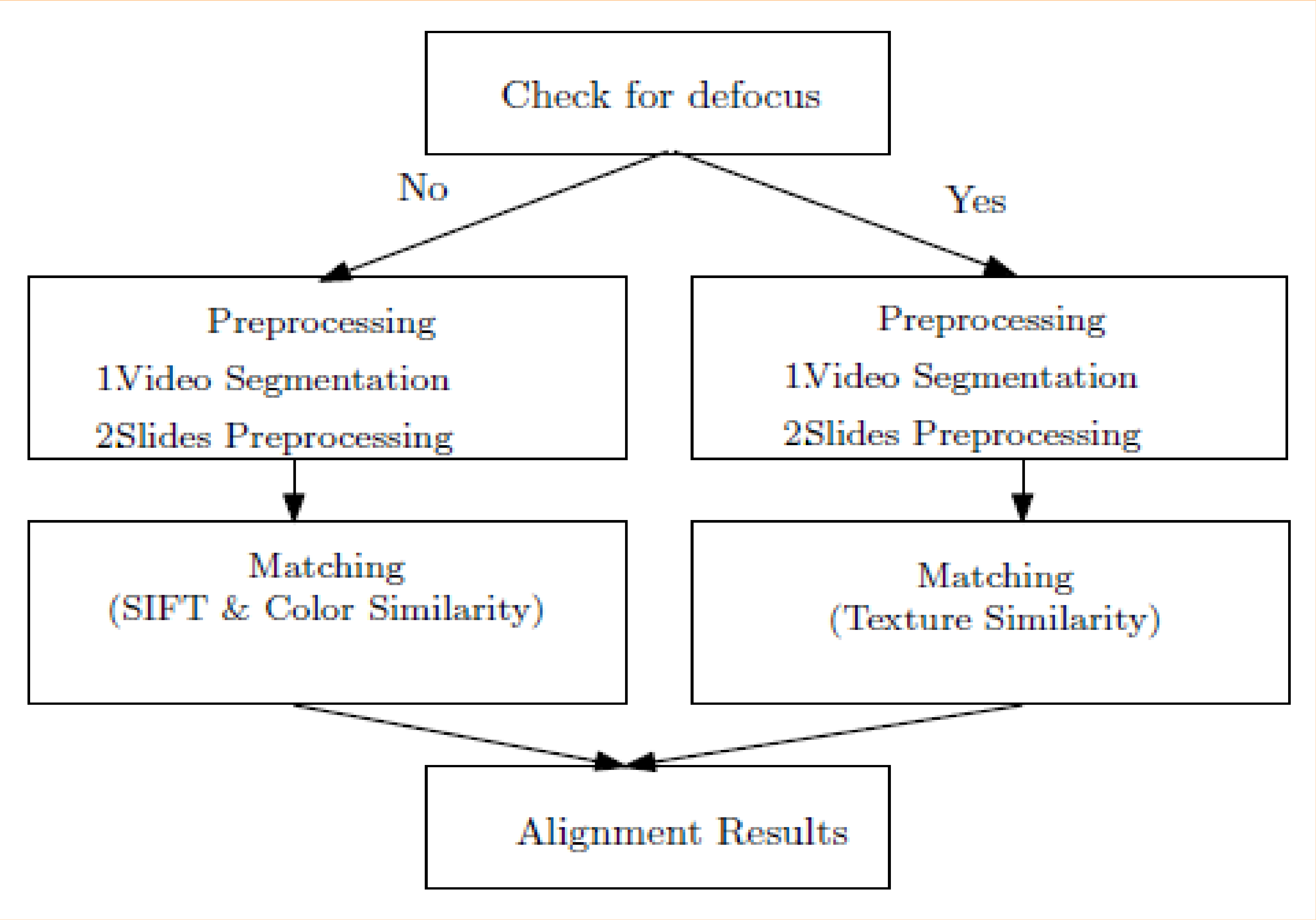


Animated slide sequence example

Descriptors: SIFT keypoints' disadvantages (3)

- Cannot handle animated slides well
- Fails to match slides in the defocused videos because of the distortion in the text region

Flow of the alignment algorithm



Check for defocus

- Only happens in videos of style 3
- For videos of other styles the slide generally will be in focus because the camera can move
- For the slide region in frame *A* and the corresponding slide image *B*, resize them and compute their gradients.
- Frame *A* is considered blurred if

$$\frac{N_A}{N_B} \geqslant \tau$$

where N_A and N_B are the number of nonzero elements in gradients

(c) Style 3: slide in background

Video preprocessing: segmentation for videos in focus

Use grayscale histogram with chi-square distance method

- 64 bin gray level histograms of frame images

- compute chi-square histogram difference

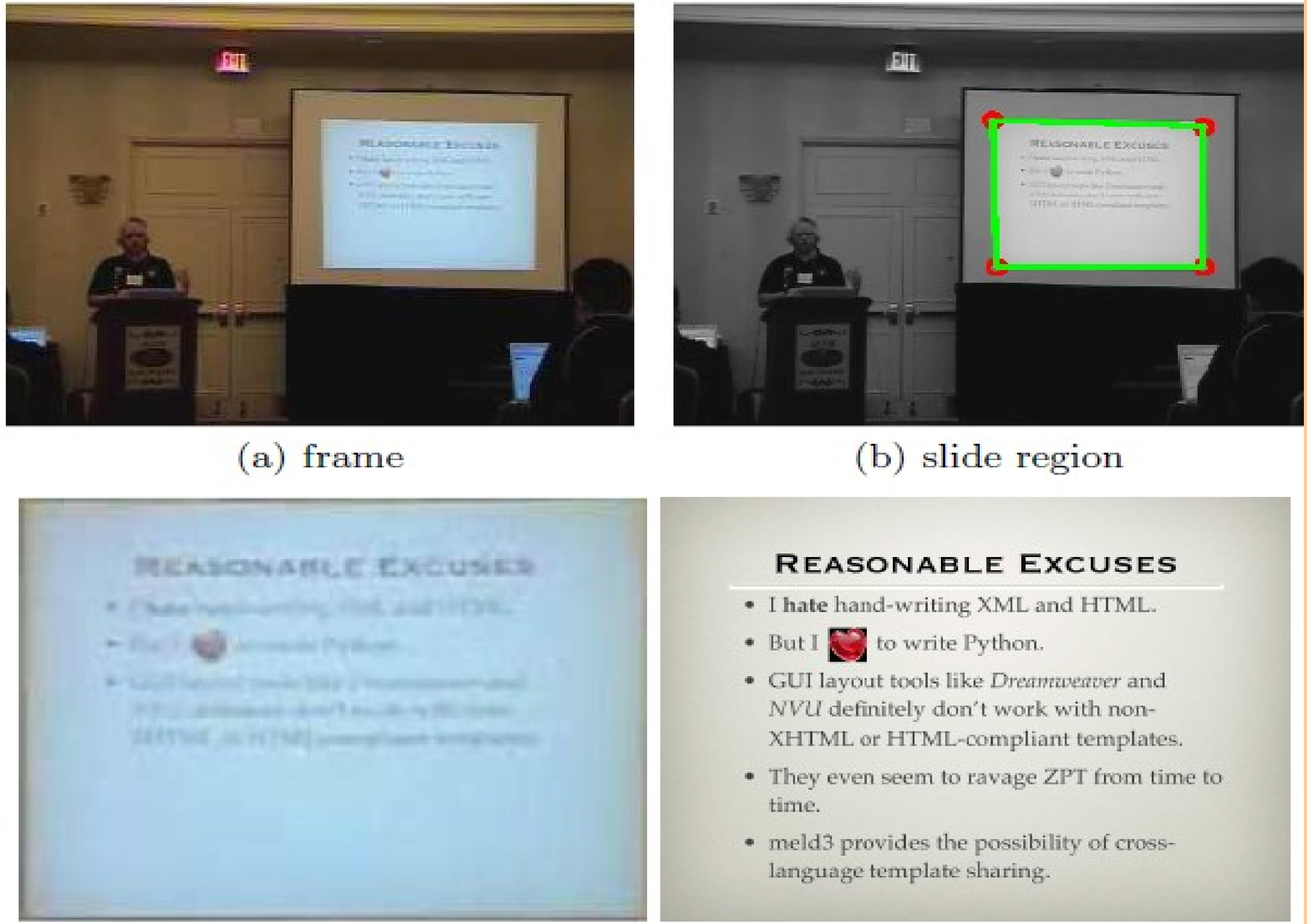
$$fd_{chi} = \begin{cases} \frac{1}{N^2} \sum_i \frac{(h_1[i]-h_2[i])^2}{h_2[i]}, & h_2[i] \neq 0 \\ \frac{1}{N^2} \sum_i \frac{(h_1[i]-h_2[i])^2}{h_1[i]}, & h_2[i] = 0 \end{cases}$$

where *h1* and *h2* are the grayscale histograms for two frames and *N* is the number of pixels in a frame

Video preprocessing: segmentation for defocused videos

- Obtain the unwarped slide region from the frame
- Since the camera and the projector are fixed, the corner points on the slide are fixed as well
- Obtain the corners of the quadrilateral using Hough transform ($\theta = 1$, $\rho = 3$)
- Compute the homography *H* to extract the slide region from the frames
- Unwarp the slide by undoing the projection *H*

Video preprocessing: segmentation for defocused videos



(a) frame

(b) slide region

(c) extracted slide region and slide image

Video preprocessing: segmentation for defocused videos

- Apply the Canny edge detection to an image G
- For two successive slide region images *A* and *B* compute
  - *a*, *b* the number of white pixels in *A* and *B* respectively
  - *a′* the number of white pixels in *A* whose corresponding pixels in *B* are also white
  - *b′* the number of white pixels in *B* whose corresponding pixels in *A* are also white
- The similarity between *A* and *B* is $M_{AB} = min(\frac{a'}{a}, \frac{b'}{b})$
- If similarity < 0.75, consider it a slide transition

Slides preprocessing: animated slide removal

- SIFT keypoints are detected for all the slides
- Lowe's nearest neighbor matching algorithm is used to get putative correspondences
  - P_B is a keypoint in image B
  - P_{A1} and P_{A2} are the 1st and 2nd nearest neighbors of P_B in image A
  - P_{A1} is a match to P_B if $\frac{d(P_{A1}, P_B)}{d(P_{A2}, P_B)} < distRatio$
  - Matched keypoints in images A and B are M_A and M_B
- RANSAC is used to get the homography between two images by solving $M_B = H*M_A$
- A is considered a part of B if $\frac{N_{inlier}}{N_{matches}} > matchRatio$

Matching Algorithm: First Phase

- Extract a keyframe *F* from each video segment
- Compute the similarities for each keyframe *F* with all electronic slide images *S*
  - Given a keyframe *F* and an electronic slide image *E* the keypoints of *F* and *E* are P_F and P_E respectively
  - Using the nearest neighbor matching algorithm find the putative correspondences M_F and M_E
  - Use RANSAC to find the true correspondences by imposing a homography on M_F and M_E

Matching Algorithm: Second Phase (1)

- Extract the slide region in the frame using the homography derived in the first phase
- For the corresponding regions, compute the color histograms and measure the similarity
  - Divide the image into 3x3 grid
  - Weigh each cell by the following filter $w = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
  - In each region compute the color histogram
  - Compute the similarity using the Bhattacharyya distance: if the color distribution is *p* and *q* and *X* is the color domain, then the distance is given by

$$BC(p,q) = \sum_{x \in X} \sqrt{p(x)q(x)}$$

Matching Algorithm: Second Phase (2)

- The color similarity between the two images A and B

$$C(A,B)=\frac{1}{\sum_{i=1}^{3}\sum_{j=1}^{3}w_{ij}}\sum_{i=1}^{3}\sum_{j=1}^{3}w_{ij}BC(i,j)$$

- where w_{ij} denotes the weight at (*i*, *j*) in the filter and BC(*i*, *j*) denotes the color similarity between the region A(*i*, *j*) and B(*i*, *j*)

Matching Algorithm

- The similarity between two images A and B is computed

$$Similarity(A, B) = N_{AB} * N^{C(A,B)}$$

- N_{AB} is the number of inliers in matched SIFT keypoints
- N is the maximum number of inliers between one frame F and each slide $S = \{s_1, \ldots, s_m\}$
- $C(A, B)$ is the color similarity

Matching Algorithm

- A hidden Markov model (HMM) is adopted to increase accuracy
- Temporal locality for the order of showing slides
  - If a frame F_t is showing slide S_i, there is a high probability that F_{t+1} will show either slide S_i, S_{i+1} or S_{i-1}
  - Nearby slides get higher probability (0.2)
  - All other slides get equal probability (equally divided for the remaining)

Matching algorithm for a defocused video

- Use the layout information
- For each segment of the presentation video, the last slide region image is chosen to match with all the electronic slides.
- The similarity is measured using the Hausdorff distance

Experiment and Results

Table 1. Data Set

Test Set	Description	Duration	Slides	Style
1	MLMI'07[1]	29min	63	2
2	MLMI'07	25min	28	2
3	MLMI'07	17min	13	2
4	CMU lecture	63min	39	1
5	Plone Symposium'06[2]	37min	14	3
6	Plone Symposium'06	39min	21	3

Table 2. Animated slides removal

Test Set	Total Slides	Animated Slides	Removed Slides
1	63	36	37
2	28	6	6
3	13	0	0
4	39	1	1
5	14	0	0
6	21	0	0

Experiment and Results

Table 3. Slide transition detection

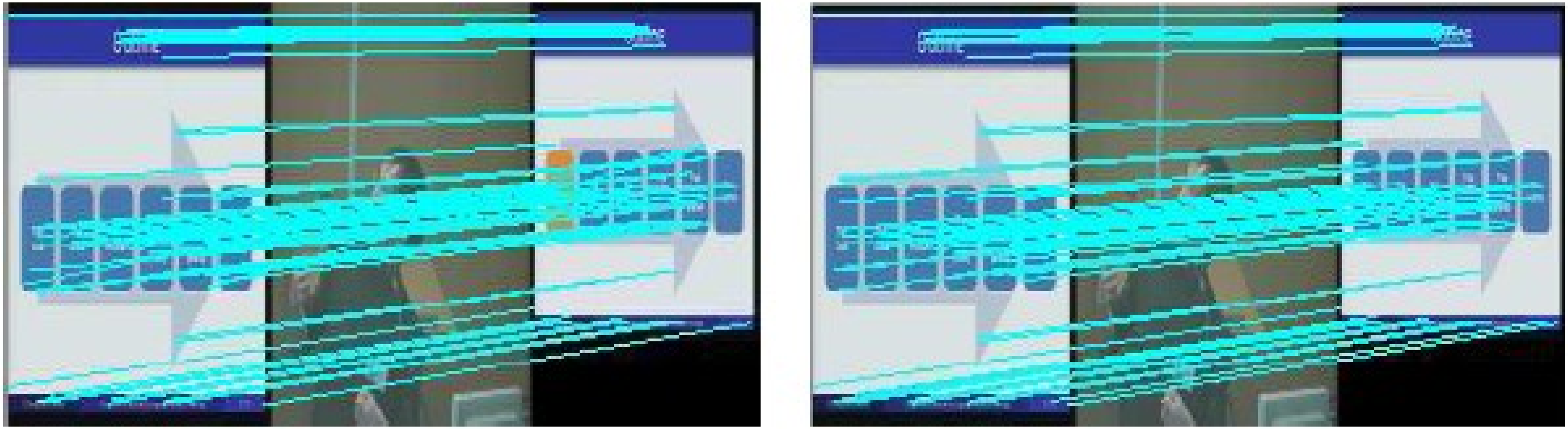
Test set	Transition	Detected Transition	
		total	correct
5	14	15	14
6	21	70	21

Table 4. Accuracy of Alignment using SIFT & color

Test set	w/o SP		with SP	
	S	S & C	S	S & C
1	78.2%	95.2%	84.4%	97.7%
2	78%	81.8%	83%	98.1%
3	38.5%	54.5%	38.5%	54.5%
4	27.5%	85.2%	28.8%	90.6%

Table 4: SP (slide preprocessing), S (SIFT), C (color). The accuracy is the ratio of correctly aligned video segments and total video segments

Alignment Results



(a) wrong alignment with SIFT only method (b) correct alignment with SIFT and color information

Results for defocused videos

(a) wrong alignment

(b) correct alignment

Results for defocused videos

Table 5. Comparison of Alignment Accuracy using Color and Texture

Test set	Transition	Error	
		Color	Texture
5	15	14	6
6	70	70	35

- Texture features work when SIFT fails
- They are also better than color features
- Yet, the error rate is still 45%
- Can only deal with fixed camera
- Suffer from occlusion problems

Questions?