Boundaries, Motion and Automatic Search at The University of Iowa

David Eichmann,[1,2] and Dong-Jun Park,[2]

[1]School of Library and Information Science
[2]Computer Science Department
The University of Iowa
david-eichmann@uiowa.edu

Abstract

Shot Boundary Detection

Approaches:

- UIowa05SB01 – by frame histogram similarity
- UIowa05SB02 – by frame pixel distance similarity
- UIowa05SB03 – by frame histogram * distance (product)
- UIowa05SB04 – by frame HSB similarity
- UIowa05SB05 – by frame pixel distance & HSB
- UIowa05SB06 – by frame product & HSB

Failure to remove error induced in last year's results makes any conclusions problematic.

Low Level Feature Extraction

Approach – sliding region window with pixel distance similarity, aggregated with run length threshold:

- UIowa05LF01 – run length of 5 frames, window range of 5 pixels +- telltale location
- UIowa05LF02 – run length of 5 frames, window range of 10 pixels +- telltale location
- UIowa05LF03 – run length of 10 frames, window range of 5 pixels +- telltale location
- UIowa05LF04 – run length of 10 frames, window range of 10 pixels +- telltale location

No distinction in performance for task as defined. False negatives are typically fast pans/tilts resulting in window overruns. Zoom logic is typically the cause of false positives for pans and tilts. We definitely need to rework our zoom logic and address coarse motion.

(Automatic) Search

Approach – fully automatic search involving two different architectures, one TDT-derived (UIowa05ASxx) and one SVM-based (uiDJx):

- UIowa05AS01 – text-only, named entity vector matches against provided sample
- UIowa05AS02 – text-only, named entity vector matches against provided sample
- UIowa05AS03 – key frame pixel distance similarity with provided samples
- uiDJ1 : color information only, based on HSB color space and is calculated from average hue, saturation and brightness values from given image
- uiDJ2 : edge information only, based on Canny's edge detection algorithm with a global edge ratio
- uiDJ3 : texture information only, based on Gray Level Cooccurrence Matrices (GLCM), which provides angular second moment, contrast, correlation, inverse difference moment and entropy measures

Text-only runs much more effective than those based upon key frames. Text-only performance on TRECVID-style topics quite variable. Next steps are to look to more frames in image-based comparison schemes.

1 – Shot Boundary Detection

As described for previous workshops [3, 4], our shot boundary work was based upon three core techniques: histogram similarity, aggregate pixel distance and aggregate edge distance. Our composite HSB technique first does a histogram-based cut detection and then overlays that with an averaged HSB gradual detection, with graduals trumping any contained cuts.

Our official runs for this year (Table 1) still exhibit performance issues similar to those reported last year. Our assumption last year was that test logic accidently left on for the evaluation runs had deleted the majority of the boundary declarations. Given the only change from last year to this was attention to the disabling of this flag, it appears instead that integration of our new gradual transition logic is actually the root cause of our performance degradation or that from 2003 (as shown in Table 2). While the transition logic has improved gradual precision, frame recall and frame precision, it's clearly damaging overall....

Table 1: Shot Boundary Task, Overall Results 2005

Run	Method	All		Cuts		Gradual			
		Rec	Prec	Rec	Prec	Rec	Prec	F-Rec	F-Prec
UIowa05SB01	histogram	0.097	0.166	0.124	0.164	0.016	0.241	0.364	0.615
UIowa05SB02	distance	0.154	0.306	0.207	0.307	0.000	0.000	0.000	0.000
UIowa05SB03	product	0.274	0.256	0.355	0.261	0.039	0.160	0.275	0.653
UIowa05SB04	hsb	0.055	0.232	0.010	0.256	0.185	0.228	0.548	0.717
UIowa05SB05	distance & hsb	0.192	0.318	0.206	0.299	0.152	0.418	0.573	0.790
UIowa05SB06	product & hsb	0.289	0.273	0.348	0.265	0.117	0.369	0.460	0.841

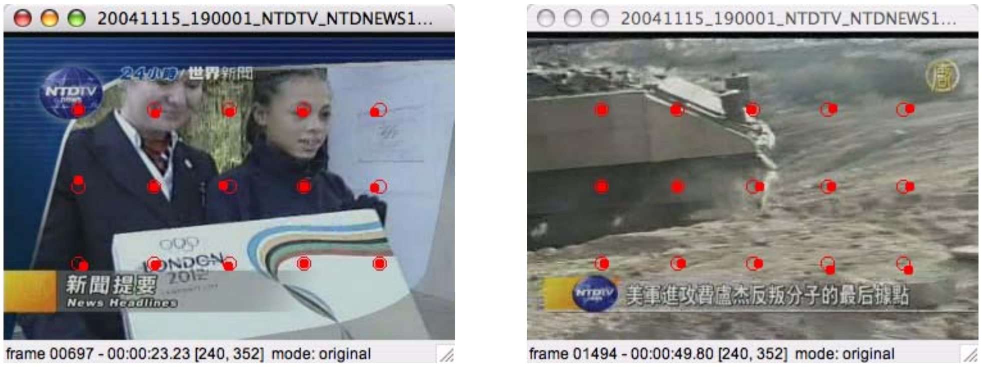
Table 2: Shot Boundary Retrospective

Year	Method	All		Cuts		Gradual			
		Rec	Prec	Rec	Prec	Rec	Prec	F-Rec	F-Prec
2003	histogram	0.445	0.804	0.554	0.937	0.178	0.389	0.234	0.960
2004		0.089	0.118	0.120	0.117	0.024	0.123	0.324	0.649
2005		0.097	0.166	0.124	0.164	0.016	0.241	0.364	0.615
2003	distance	0.607	0.855	0.835	0.963	0.051	0.158	0.178	0.826
2004		0.083	0.119	0.121	0.120	0.003	0.109	0.400	0.328
2005		0.154	0.306	0.207	0.307	0.000	0.000	0.000	0.000
2003	product	0.722	0.785	0.893	0.976	0.306	0.330	0.300	0.938
2004[a]		-	-	-	-	-	-	-	-
2005		0.274	0.256	0.355	0.261	0.039	0.160	0.275	0.653

[a] No pure product run was submitted for 2004

2 – Low-Level Feature Extraction

The detection of low level camera motion has substantial potential utility as a building block in the construction of higher level features. Given this, our primary interest in the detection of camera motion is not the declaration of presence or absence of a given motion in a given shot (i.e., the task as defined this year), but rather the recognition of


20041115_190001_NTDTV_NTDNEWS1...
24小時/世界新聞
NTDTV
新聞提要
News Headlines
LONDON 2012
frame 00697 - 00:00:23.23 [240, 352] mode: original
20041115_190001_NTDTV_NTDNEWS1...
NTDTV
美軍進攻費盧杰反叛分子的最后據點
frame 01494 - 00:00:49.80 [240, 352] mode: original

(a): A zoom in with a tilt down

(b): A right pan

Figure 1: Sample frames with overlaid telltales

fine-grained motion of arbitrary duration, even over 1-2 frames. Furthermore, we are interested in computationally efficient techniques that can function well as a node in a processing pipeline in near-real-time.

First a comparison representation is selected - the original image was used for our official runs, but we have subsequently experimented with gray-scale and edge recognition outputs. We define a grid of monitoring points within the frame (3 by 3 in our official runs) and an evaluation region-of-interest (5 by 5 pixels for our official runs). Then for each new frame, we measure the cumulative pixel distance (as defined in our previous work in shot boundary detection) of each monitoring point's ROI against the preceding frame's evaluation window, 5 by 5 and 10 by 10 pixels for our official runs. The coordinate of the highest match provides a motion vector, both a direction of motion and a measure of that motion and the similarity serves as a confidence measure for that motion.

The resulting array of motion vectors can be used for a number of motion recognition tasks. For our purposes this year, a majority of vectors exhibiting non-zero horizontal components with the same sign implies a pan in that direction and a majority with non-zero vertical components with the same sign a tilt in that direction. Our zoom logic for the official runs is fairly preliminary, involving opposing motion in left- and right-most columns and opposing motion in top- and bottom-most rows.

To perform the task as defined, declaration of individual motion elements are aggregated into run length vectors. Our official runs set two declaration thresholds, 5 and 10 contiguous frames.

For purposes of visualization during development, we decorate the video frame with telltales[*] indicating localized tracking direction. Figure 1 shows two representative frames and their corresponding telltales. These telltales can be quite useful in debugging preliminary designs for motion detectors. Figure 1a, for example, involves tilt down composed with an inward zoom, but one that is centered on the middle of the card seen in the lower portion of the frame. Note that the right column is tracking this well, but the relatively smooth blue surfaces of the uniforms results in somewhat random directional jitter. Furthermore, the overlay graphics causes the upper-left telltale to be completely ineffective.

Figure 1b involves a right pan that is tracking a military vehicle. Note in this case that while the majority of telltales are detecting this pan, those overlaying the vehicle show no motion (as would be expected). Difficulties arise in these types of shots when tracking is only approximate – relative motion of the tracked object can be in virtually any direction, and if the percentage of frame occupied by the object of interest for the shot is substantial, it can result in misses or false positives.

Table 3 shows the results for our official runs. Note that our run length and range settings produce no distinction in performance. We are distinctly skewed towards recall with these settings, but our modest precision results are more likely colored more by our poor zoom recognition being translated into pan and tilt false alarms. Observation of telltale

[*] A telltale in sailing is a small piece of (usually) yarn attached by one end to a sail. It is used to judge direction and regularity of the air flow while trimming sail.

and declaration activity bears this out. Much of our false negative rate is the result of our overly-conservative range settings. Newswire footage in this corpus has a substantial number of very fast pans and tilts. These are not recognized because the inter-frame pan distance is greater than the range settings used.

Table 3: Low-Level Feature Extraction

Run	Run Length	Range	Pan		Tilt		Zoom		Mean	
			Prec	Recall	Prec	Recall	Prec	Recall	Prec	Recall
UIowa05LF01	5	5	0.172	0.513	0.064	0.414	0.016	0.053	0.084	0.327
UIowa05LF02	5	10	0.172	0.513	0.064	0.414	0.016	0.053	0.084	0.327
UIowa05LF03	10	5	0.172	0.513	0.064	0.414	0.016	0.053	0.084	0.327
UIowa05LF04	10	10	0.172	0.513	0.064	0.414	0.016	0.053	0.084	0.327

Our future work in this area will include experimentation with a variety of motion declaration schemes, telltale declaration patterns, and a hybrid ranging scheme that attempts to balance out detection of both coarse and fine grained camera motion while maintaining a reasonable evaluation cost [1, 5].

3 – Automatic Search

This is our first year in this task. As a small group, we've chosen to build a fully automatic search engine. Our primary design goal is an engine that can be composed from a potentially large number of comparators, detectors and filters. We have an established library of entity recognition components derived from our work in TREC and TDT. Our text-only official runs (UIowa05AS01 and 02) involved named entity recognition of the topic definition and the ASR for the sample video. These entity vectors are then used to generate our standard entity similarity measure against named entity vectors extracted from the ASR output from the test corpus. Given the similarity in corpora, our runs are hence a projection of TDT-style search techniques onto TRECVID-style topic definitions.

Performance for the text-only runs varies widely across topics, as shown in Table 4. We plan on establishing the nature of recurring failure modes for poor performing topics and the connection to TRECVID-style topics as compared to TDT-style topics.

Our video+ASR UIowa05AS03 run performs whole-image/frame comparisons from the candidate pool to the provided key frames for the test corpus. We see this as a preliminary solution. Subsequent work has involved topic enhancement through Web image meta search, skin/face recognition and texture comparison components.

We submitted 3 runs from using image features (uiDI1, 2 and 3). Each run uses a different image feature.

- uiDJ1 : color information only, based on HSB color space and is calculated from average hue, saturation and brightness values from given image
- uiDJ2 : edge information only, based on Canny's edge detection algorithm with a global edge ratio
- uiDJ3 : texture information only, based on Gray Level Cooccurrence Matrices (GLCM), which provides angular second moment, contrast, correlation, inverse difference moment and entropy measures

This automatic search system configuration is based on the previous year's shot boundary and feature extraction task system. Rather than producing the above features from smaller blocks, we opted to use the global value, calculating the metric for the full frame. This, in turn, allowed faster processing time. The simple Euclidean distance for each feature was used to estimate the similarity between images.

Table 5 shows interpolated recall precision for all runs and Table 6 precision at n shots.

References

[1] Dietterich, T. G., "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting and Randomization," *Machine Learning,* v. 40, no. 2, 2000, p. 139-157.

Table 4: Automatic Search Results

Topic	Unq. Hits	UIowa05AS01					UIowa05AS02					UIowa05AS03				
		Ave. Prec	Hits at Depth				Ave. Prec	Hits at Depth				Ave. Prec	Hits at Depth			
			10	30	100	1000		10	30	100	1000		10	30	100	1000
149	116	0.033	3	6	13	23	0.038	3	6	15	32	0.000	0	0	1	1
150	13	0.004	0	0	2	2	0.005	0	0	2	2	0.000	0	0	0	0
151	301	0.023	1	2	8	81	0.027	1	4	8	84	0.000	0	0	0	0
152	498	0.158	1	9	35	254	0.164	1	9	37	259	0.000	0	0	0	0
153	42	0.089	5	5	12	14	0.102	5	5	10	14	0.000	0	0	0	1
154	93	0.024	0	1	7	40	0.039	1	2	8	48	0.000	0	0	0	0
155	54	0.000	0	0	0	2	0.000	0	0	0	2	0.000	0	0	0	0
156	55	0.000	0	0	0	4	0.000	0	0	1	1	0.000	0	0	0	0
157	470	0.004	0	3	3	42	0.004	0	1	2	42	0.000	0	0	0	1
158	63	0.001	0	0	1	4	0.001	0	0	2	4	0.000	0	0	0	0
159	29	0.000	0	0	0	1	0.000	0	0	1	2	0.000	0	0	0	0
160	169	0.001	0	0	0	15	0.001	0	0	0	15	0.000	0	0	0	0
161	1245	0.004	0	3	9	58	0.004	0	3	9	57	0.000	0	0	0	0
162	385	0.000	0	0	3	10	0.000	0	0	1	10	0.000	0	0	0	0
163	1160	0.023	2	6	26	117	0.025	3	11	26	113	0.000	0	0	0	0
164	214	0.010	0	3	12	27	0.013	0	2	17	33	0.000	0	0	0	0
165	254	0.002	0	0	2	17	0.003	0	1	3	23	0.000	0	0	0	0
166	253	0.001	1	1	1	11	0.001	1	1	2	11	0.000	0	1	1	3
167	19	0.000	0	0	0	0	0.001	0	3	3	17	0.000	0	0	0	0
168	1087	0.000	0	1	3	16	0.001	0	3	3	17	0.001	1	1	1	1
169	493	0.005	1	1	4	47	0.005	1	1	5	47	0.000	0	0	0	1
170	543	0.001	0	0	2	26	0.001	0	0	1	25	0.000	0	0	1	1
171	49	0.040	1	2	9	13	0.048	1	5	9	17	0.000	0	0	0	0
172	790	0.000	0	0	1	11	0.000	0	0	2	9	0.004	1	5	13	21

[2] Eichmann, D., “Ontology-Based Information Fusion,” *Workshop on Real-Time Intelligent User Interfaces for Decision Support and Information Visualization, 1998 International Conference on Intelligent User Interfaces*, San Francisco, CA, January 6-9, 1998.

[3] Eichmann, D and D. J. Park, “Experiments in Boundary Recognition at the University of Iowa,” *Proceedings of the 2003 TRECVID Workshop.*

[4] Eichmann, D and D. J. Park, “Boundary and Feature Recognition at the University of Iowa,” *Proceedings of the 2004 TRECVID Workshop.*

[5] Fiscus, J., “A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER),” *Proc. IEEE ASRU Workshop,* p. 347-352, Santa Barbara, CA, 1997

[6] Zabih, R., Miller, J and Mai, K., “A feature-based algorithm for detecting and classifying scene breaks,”

Table 5: Interpolated Recall Precision

Int. Recall	AS01	AS02	AS03	DJ1	DJ2	DJ3
0.0	0.1963	0.2310	0.0416	0.0945	0.0670	0.1582
0.0	0.0659	0.0778	0.0000	0.0011	0.0020	0.0000
0.2	0.0285	0.0301	0.0000	0.0002	0.0000	0.0000
0.3	0.0193	0.0201	0.0000	0.0000	0.0000	0.0000
0.4	0.0145	0.0155	0.0000	0.0000	0.0000	0.0000
0.5	0.0109	0.0133	0.0000	0.0000	0.0000	0.0000
0.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.7	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.9	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 6: Precision at N Shots

# Shots	AS01	AS02	AS03	DJ1	DJ2	DJ3
5	0.0750	0.1000	0.0174	0.0667	0.0083	0.0583
10	0.0625	0.0708	0.0087	0.0500	0.0125	0.0417
15	0.0639	0.0611	0.0087	0.0500	0.0194	0.0361
20	0.0604	0.0646	0.0065	0.0458	0.0229	0.0333
30	0.0611	0.0750	0.0101	0.0444	0.0208	0.0264
100	0.0637	0.0683	0.0074	0.0300	0.0271	0.0221
200	0.0521	0.0535	0.0048	0.0181	0.0204	0.0150
500	0.0421	0.0431	0.0024	0.0126	0.0168	0.0087
1000	0.0348	0.0361	0.0013	0.0094	0.0135	0.0057

Third International Multimedia Conference and Exhibition, Multimedia Systems, pages 189-200, San Francisco, California 1995