Experiments in Boundary Recognition 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

The University of Iowa participated in the shot boundary detection and story segmentation tracks of TRECVID-2003.

1 – Shot Boundary Detection

Our system for this task was based upon an initial hypothesis that a relatively small number of 'basic' metrics could be used in fairly simple combination to construct metrics that performed well on complicated data. This hypothesis was derived from similar work in speech recognition (e.g., ROVER [3]), and recent work with ensembles in machine learning (e.g., [1]).

We opted to work solely on localized (frame pair-wise) measures, seeking to establish just what they could contributed in a pure environment to the task. This clearly has significant impact on the ability to detect gradual transitions, as will be discussed later, but it was interesting to discover that our composite metrics could perform as well as they did (e.g., 30% recall & 33% precision for our product metric) for an inherently frame sequence-driven task.

Our official runs involved the following approaches for the defined task:

Basic Methods

Color Histogram Similarity. This is a simple frequency count histogram measure, with 512 bins. Pixels are mapped to a bin by extracting distinct 8-bit RGB values, right-shifting each value 5 bits, and recomposing the remaining 3 bits into a 9-bit color value, which is used as the bin index. This not only reduces storage and computation requirements, it also has the desirable side-effect of damping out minor color variations in the video due to recording or encoding artifacts.

Frame Color Distance Similarity. This measure uses as input two 60-by-60 thumbnails of the candidate images and computes the cumulative cosine-vector distance in RGB color space of corresponding pairs of pixels. The cumulative distance is then averaged over the area of the thumbnails. Thumbnails rather than full images are used, as with the histograms, to reduce computational requirements, and as a side-effect to suppress fine-grained detail variations between images.

Frame Edge Distance Similarity. This method looks at the changes in the edges detected in adjacent frames as introduced by Zabih, Miller, and Mai [4]. Each frame is transformed into a grayscale image and has applied the Sobel operator to detect edges. The method then computes the percentage of edges that enter and exit between two adjacent frames. Zabih, Miller, and Mai suggests to align the images to compensate motion, but we decided not to do it to lessen the burden of computation. This yielded very high values of edge changes and made this approach somewhat insensitive to shot boundary changes.

Composite Methods

Boolean Predicate of Basic Methods. This approach entails the composition of a boolean predicate of the basic metrics to form a conditioned decision. We submitted runs for two such predicates, which are listed in Table 1.

Product of Basic Methods. This metric is the arithmetic product of histogram, distance and edge similarities. Since each of these ranges over the interval [0:1] the composite does as well. This approach avoids the need to intuit a pred-

icate and does well at balancing out a strong response in a single basic metric with at least a limited response in the remaining metrics.

Development Activities

We had already done some preliminary work with shot boundary detection on a very ad hoc basis in the context of extending a Web search agent [2] to support video as well as text and image retrieval. Because of this, we were able to concentrate our development efforts on the definition of an evaluation architecture, the refactoring of the legacy code base, and the visualization and analysis of development output for tuning. Our current environment captures the thumbnail compressed-color pixel array, the edge pixel array and the histogram data for each frame as a serialized object in a relational database. As we do this processing, we also capture pair-wise frame metrics (our basic set as described above) as separate tables. This allows us to extract video sequences and their corresponding metrics for analysis and visualization. Figure 1 shows a sample plot of measures for two fragments of development video as examples of our visualization of the data. Taking this approach proved very useful in minimizing effort expended in tuning the system – indeed, the thresholds shown in Table 1 were established by iterative evaluation of the plots and the corresponding video segments using just a handful of randomly selected development videos.

Official Runs

As mentioned above, our focus for this first year of participation was establishing performance baselines for method 'primitives' and explore some simple combinations of these primitives. Subsequent work will then layer more specialized recognition logic (e.g., a flash recognizer) onto this foundation. Table 1 shows the results for our submitted runs. We had not yet done any tuning of the internal parameters for the edge metric (i.e., the size of a candidate pixel's evaluation neighborhood) by the submission deadline, and since we were not seeing a large range of response values, chose not to submit a run for this metric. We're now seeing a more interesting range of values out of this metric.

Table 1: Shot Boundary Task, Overall Results

Run	Method	Threshold	All		Cuts		Gradual			
			Rec	Prec	Rec	Prec	Rec	Prec	F-Rec	F-Prec
UIowaSB0301	histogram	0.80	0.445	0.804	0.554	0.937	0.178	0.389	0.234	0.960
UIowaSB0302	distance	0.60	0.607	0.855	0.835	0.963	0.051	0.158	0.178	0.826
UIowaSB0303	composite-1	h<0.95 & (d<0.80 | e<0.85)	0.657	0.785	0.810	0.948	0.285	0.360	0.274	0.907
UIowaSB0304	product	d*e*h<0.60	0.722	0.785	0.893	0.976	0.306	0.330	0.300	0.938
UIowaSB0305	composite-2	(h<0.82 & d<0.82) | (h<0.79 & e<0.79)	0.665	0.432	0.772	0.957	0.406	0.123	0.286	0.777

Figure 2 shows precision and recall for all videos for all submitted runs. The outliers on the perimeter of the plot are the C-SPAN videos from the test set. Given the extremely low number of transitions present in this source, recognition proved to be an all-or-nothing proposition. Figure 3 shows this broken out by cut and gradual transitions. Our decision to concentrate on pair-wise measures clearly pays off for cut transitions, as can be seen in Figure 3a, and at the clear cost shown in Figure 3b for gradual transitions. The distinction in performance between composite-1 and composite-2 for gradual transitions not only provides an explanation for the separation of composite-2 results in Figure 2, it also demonstrates the potential difficulties in the construction of predicate-base composite metrics.

Breaking results down by source, transition type and metric yields Figure 4. While the sample size is small, there seems to be a general sense that performance is more uniform for cut transitions, with fairly compact result clouds for histogram vs. distance in the basic metrics category (with distance clearly the better performer) and a general trend for

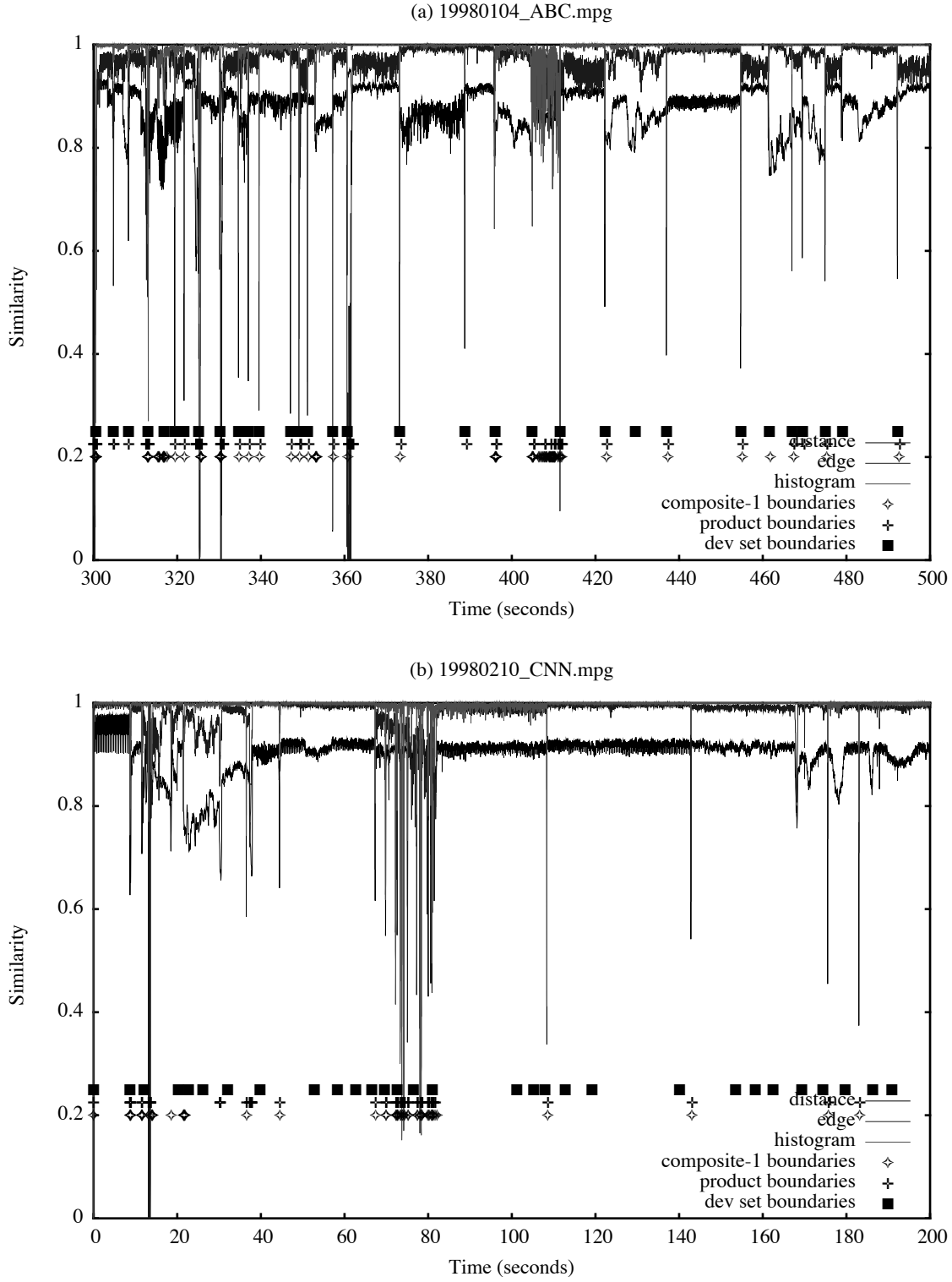

Figure 1: Sample Output Measures

the product measure to edge-out composite-1 and composite-2 in the composite metrics category. Things are a bit more diffuse and muddled for the gradual transitions. Histogram provides substantially better results than does distance, but

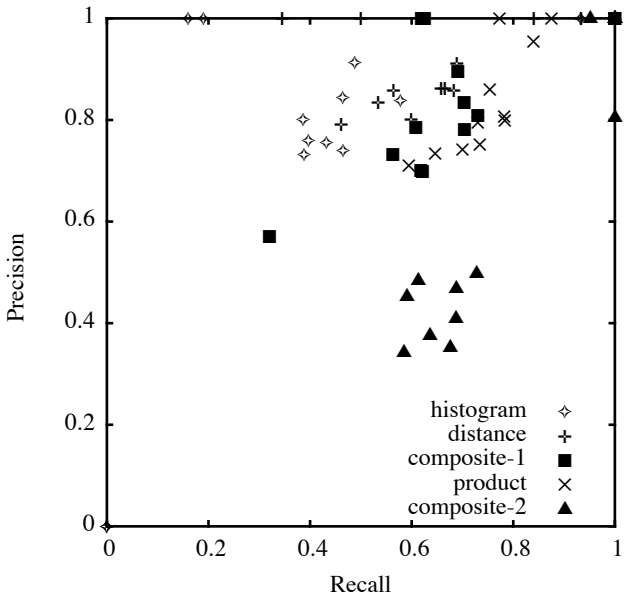

Figure 2: Shot Boundaries, Overall Results

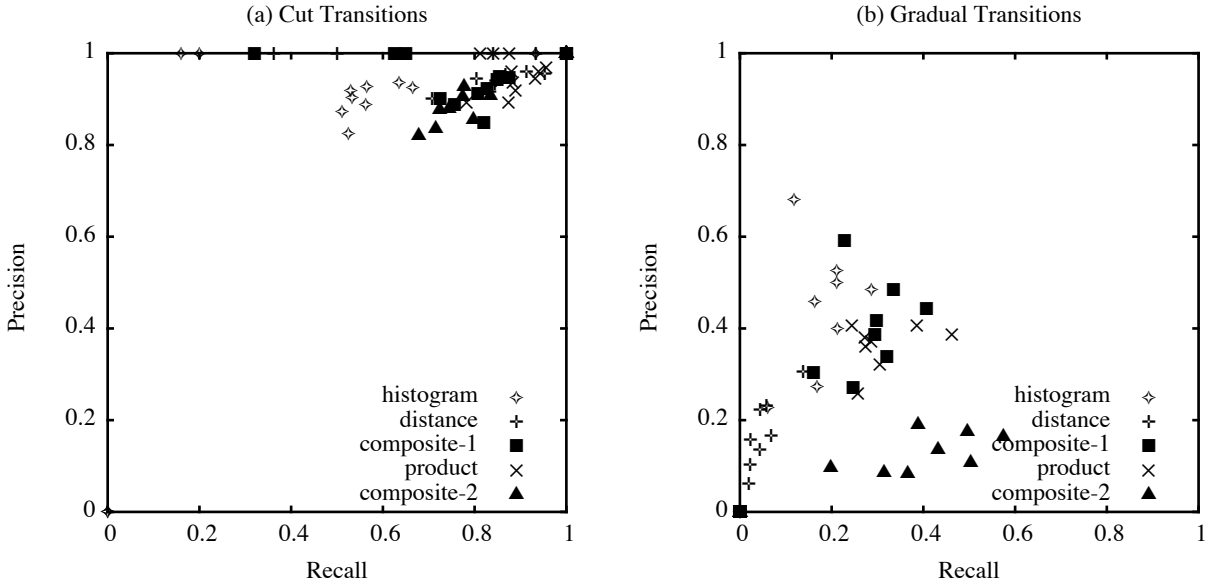

Figure 3: Shot Boundaries by Transition Type

the only real conclusion that we attempt to draw for the composite metrics is that composite-2 is probably a little too aggressive on recall, to the detriment of its precision.

Future Work on Shots

We have only begun to perform an error analysis of our results, but for the product metric, it appears that a recurring source of false alarms is camera flashes. An excellent example of this can be found in Figure 1b at time ~70 seconds through 80 seconds. The flurry of product boundary declarations is caused by a phalanx of photographers following Monica Lewinsky as she exits a building in near darkness. Each camera flash results in a false alarm.

In general, we see two clear areas to improve our performance:

Specialized event detectors. There appears to be an opportunity for identification of basic metrics that are sequence

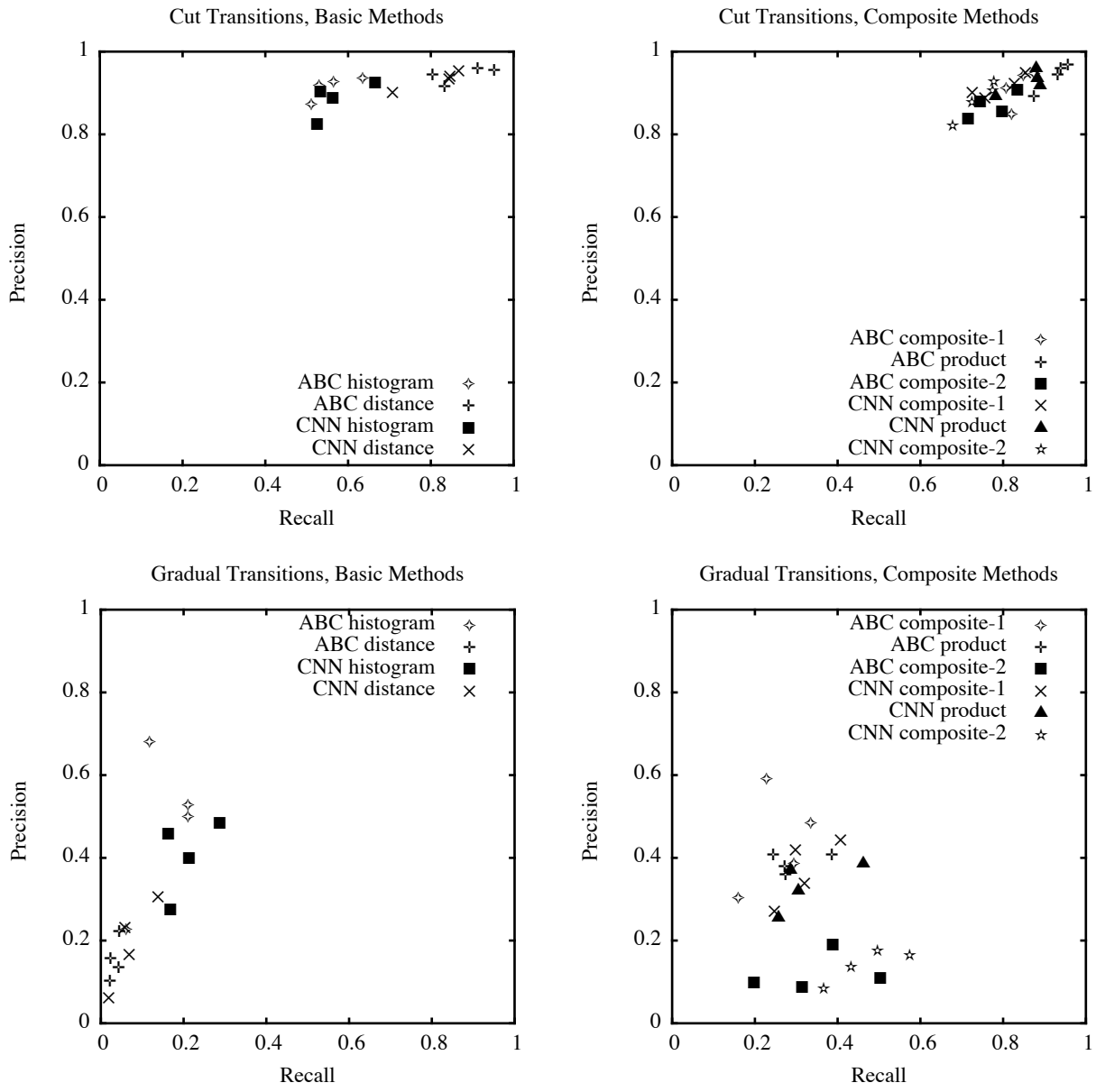

Figure 4: Shot Boundaries by Transition Type and Source

based in a manner similar to our identification of basic frame pair metrics. One such detector has already been mentioned for camera flashes. We believe that additional sequence detectors for wipes, dissolves, etc. will form a robust foundation of basic metrics for frame sequences.

Composite frame sequence metrics. We knew that gradual transitions would be problematic, given our frame pair focus. Our framework approach to metric specification and implementation will easily accommodate experiment with sequence-based metrics. We anticipate that these metrics will draw upon both categories of pair-wise metrics and upon the event detectors.

2 – Story Segmentation

Based upon our experience with TDT story segmentation, and the techniques that other TDT participants employed for that task, we focused on two aspects of the ASR data, speech pauses longer than a certain threshold and

trigger phrases (e.g., "John Smith, ABC News, Atlanta") as indications of story boundaries. For the video/audio data, we concentrated solely on the video, and used our product shot boundaries as indications of story boundaries.

Condition 1 (Video only). Our official run entailed a composite measure involving color histogram, aggregate pixel similarity and edge similarity used to define shot boundaries, and inferentially story boundaries (described in proposal 2). This resulted in a high-recall, low-precision result (as you might expect) but with noticeable differences when results are analyzed separately by source. Both precision and recall are better for CNN than for ABC using this simple technique. We considered this configuration a baseline for comparison

Condition 3 (ASR only). We submitted six runs for this condition: trigger phrase only, speech pause only and a composite measure run at two different threshold values. Trigger phrases prove to be a high precision means of identifying story boundaries, assuming that a proper set of trigger phrases have been identified. Our results show a substantial difference in recall for ABC over CNN with no sacrifice in precision. We speculate, given some additional experimentation, that the set of trigger phrases we identified using equal numbers of development videos was not sufficient to identify the full (and larger) set of trigger phrases for CNN. Speech pauses prove to be a meager source of story boundaries in ABC videos, and a substantially better source of boundaries in CNN. The composite scheme (both trigger phrases and speech pauses) results in no noticeable improvement for CNN, but substantial improvement for ABC, both in precision and recall. It is interesting to note that even though recall and precision is poor for ABC boundaries derived from speech pauses, there is sufficient information signal in the data to improve recall performance for that source over using just trigger phrases.

Condition 2 (Video & ASR). We submitted a single run for this condition which used the composite measure of condition 1 for video and the composite measure for condition 2 for ASR data. This actually proved useful in improving performance relative to the corresponding condition 3 run. Even though precision is poor for the video composite scheme, using it improves the ASR result. Effects differ for ABC and CNN, with CNN results suffering some degradation in recall but improvement in precision. ABC results little or no recall degradation for a relatively (compared to CNN) greater improvement in precision.

For news typing, we took a very simplistic approach. The first segment in every video was declared as non-news and all other segments were declared as news. Table 2 shows our overall results for all submitted runs.

Table 2: Story Boundary Task, Overall Results

Run	Text Method	Video Method	Condition	Story Boundary		News Classification	
				Rec	Prec	Rec	Prec
UIowaSS0301	trigger phrases	–	3 – ASR	0.261	0.679	0.901	0.683
UIowaSS0302	both	–	3 – ASR	0.402	0.332	0.980	0.656
UIowaSS0303	speech pauses	–	3 – ASR	0.223	0.229	0.956	0.647
UIowaSS0304	trigger phrases	–	3 – ASR	0.261	0.679	0.897	0.656
UIowaSS0305	both	–	3 – ASR	0.465	0.312	0.988	0.657
UIowaSS0306	speech pauses	–	3 – ASR	0.319	0.246	0.971	0.650
UIowaSS0307	both	product	2 – ASR & video	0.343	0.402	0.953	0.654
UIowaSS0308	–	product	1 – video	0.767	0.140	1.000	0.648

Figure 5 shows our overall results for story segmentation and news typing. Trigger phrases prove to be substantially higher in precision than speech pauses, a combination of trigger phrases and speech pauses or just shot boundaries. Trigger phrases and shot boundaries appear to provide an interesting 'bracket' for a performance trade-off between precision and recall.

Figure 6 shows results for all test videos for condition 1. Note the performance distinction between ABC and CNN, with CNN's performance noticeably better using our simplistic approach.

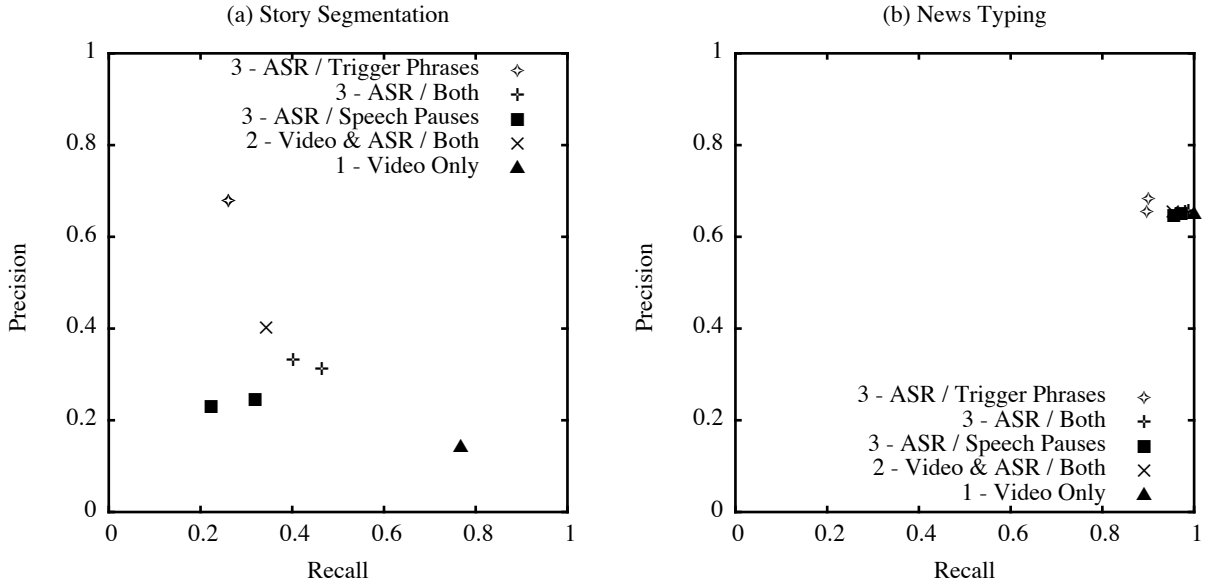

Figure 5: Story Segmentation, Overall

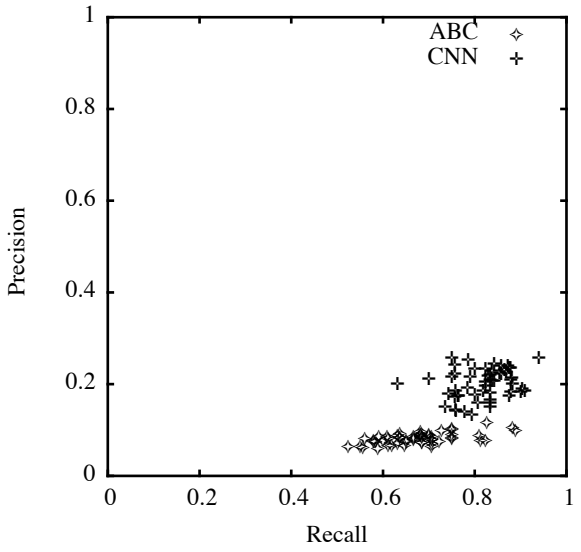

Figure 6: Story Segmentation, Condition 1, by Source

Figure 7 shows results for all test videos for condition 2. For our product metric, when combined with a condition that a shot boundary co-occurs, CNN has a slightly higher precision, but ABC's recall is generally much higher.

Figure 8 shows the block of six runs submitted for condition 3. We submitted a run for two speech pauses (1.25 and 1.50 seconds) combined with three segmentation techniques: trigger phrases (TP), speech pauses (SP) and both (TP & SP). Note that the performance spread for trigger phrases is much broader than that for speech pauses or the combined scheme. Figure 9 shows the ASR-based runs, broken out by source. Our trigger phrase scheme clearly performs well for ABC, and while precision is good for CNN, recall is poor. For ABC, combining trigger phrases with speech pauses improves performance over speech pauses alone, but seriously impacts precision over trigger phrases alone, with only a small improvement in recall. For CNN, there is more impact in lowering the speech pause threshold by a quarter second than combining speech pauses with trigger phrases. Much of the relative difference in performance between ABC and CNN appears to be potentially attributable to better coverage of the domain of trigger phrases for ABC than we have identified for CNN.

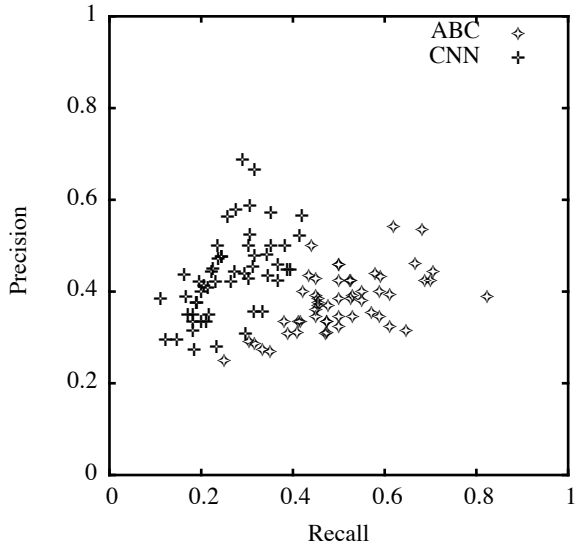

Figure 7: Story Segmentation, Condition 2, by Source

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.

[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] 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

[4] Zabih, R., Miller, J and Mai, K., "A feature-based algorithm for detecting and classifying scene breaks," *Third International Multimedia Conference and Exhibition, Multimedia Systems*, pages 189-200, San Francisco, California 1995

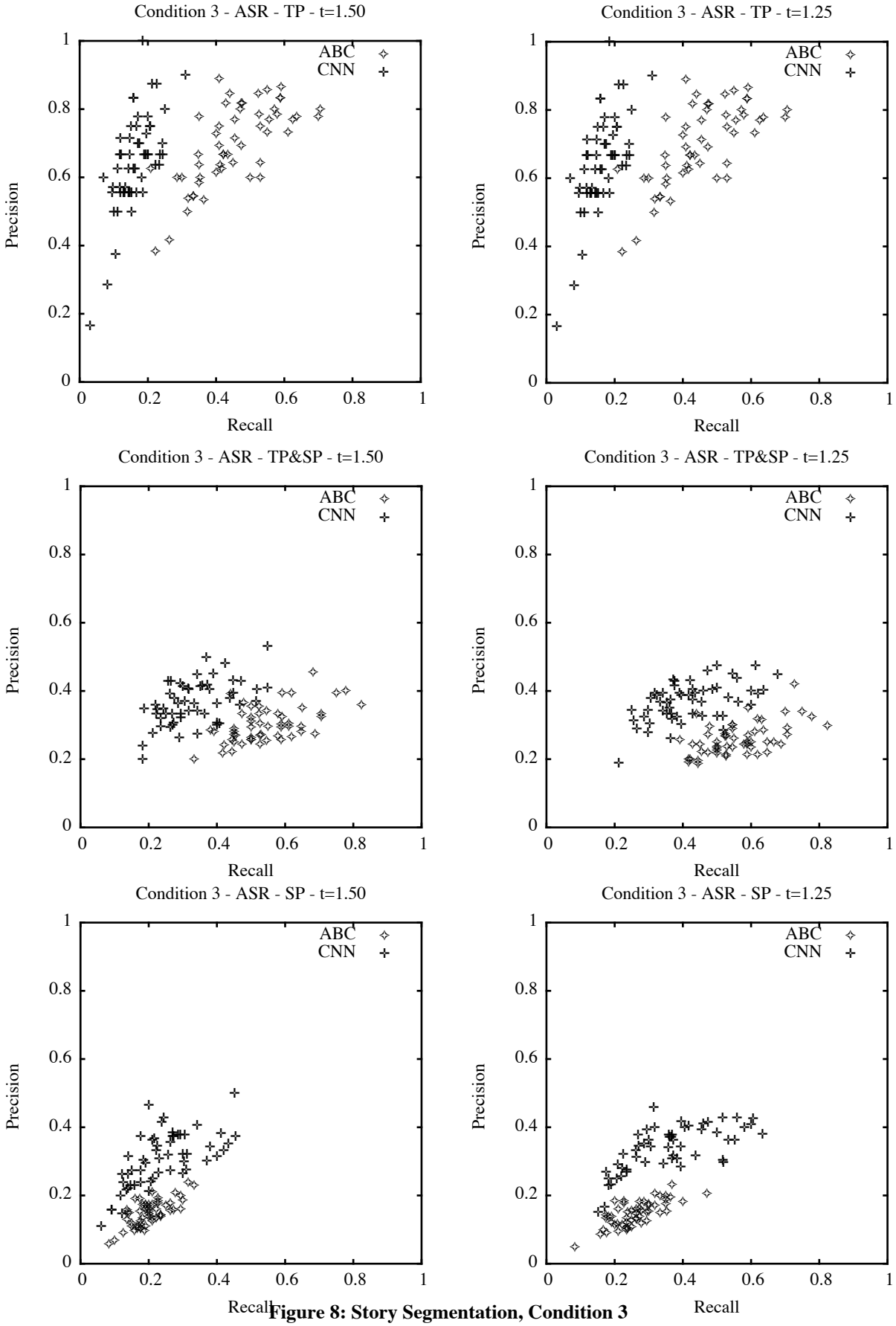

Figure 8: Story Segmentation, Condition 3

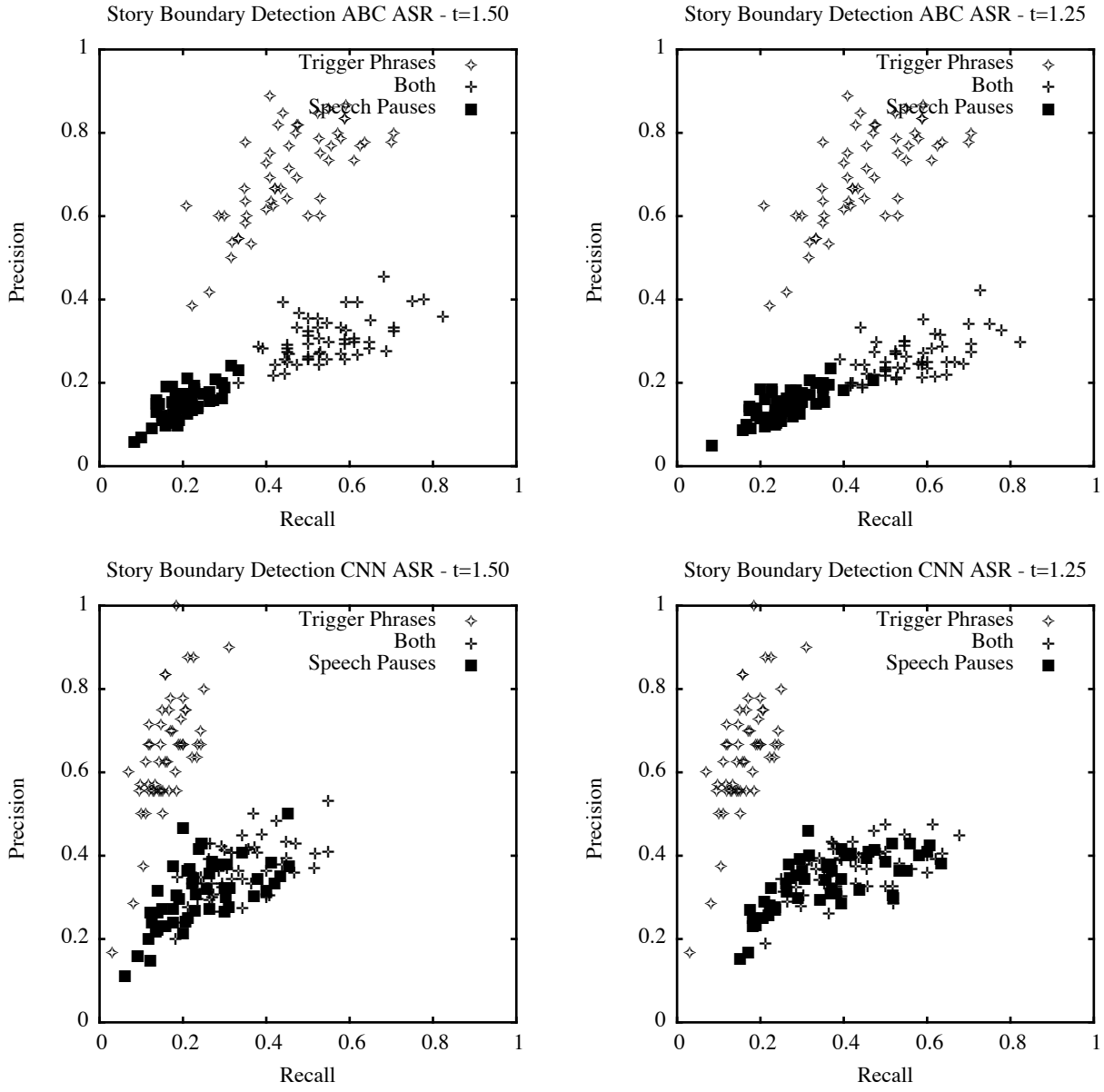

Figure 9: Story Segmentation, Condition 3, by Source