Semantic analysis of a large-scale news video archive

Ichiro Ide [†,‡] **Kazuhiro Noda** [†*] **Akira Ogawa** [†] **Shin'ichi Satoh** [‡] **Hiroshi Murase** [†]

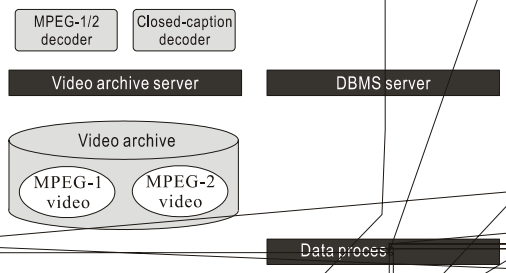
† Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan

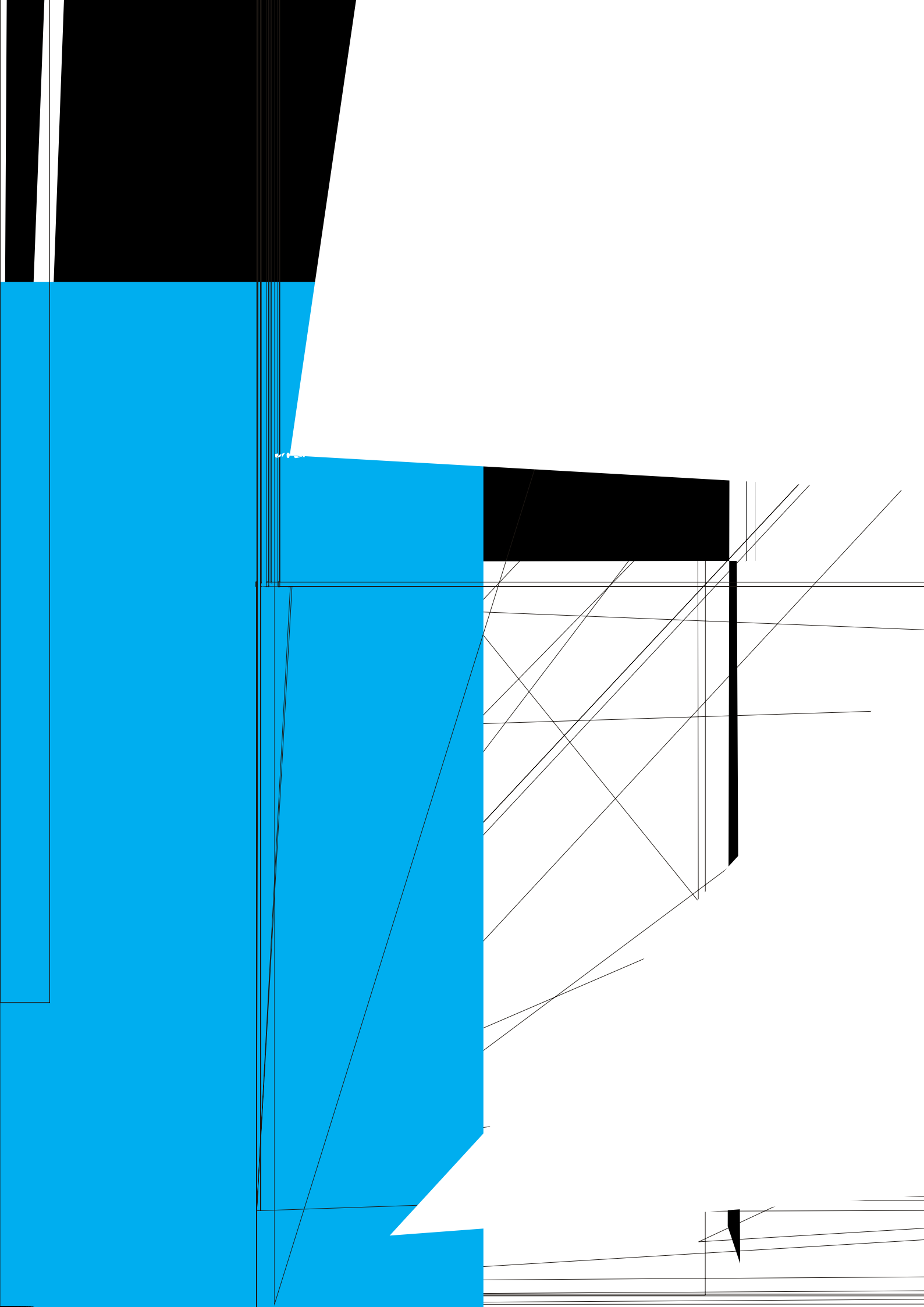
{ide, murase}@is.nagoya-u.ac.jp, {knoda, aogawa}@murase.m.is.nagoya-u.ac.jp

‡ National Institute of Informatics, Research Organization of Information and Systems
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

{ide, satoh}@nii.ac.jp

Abstract





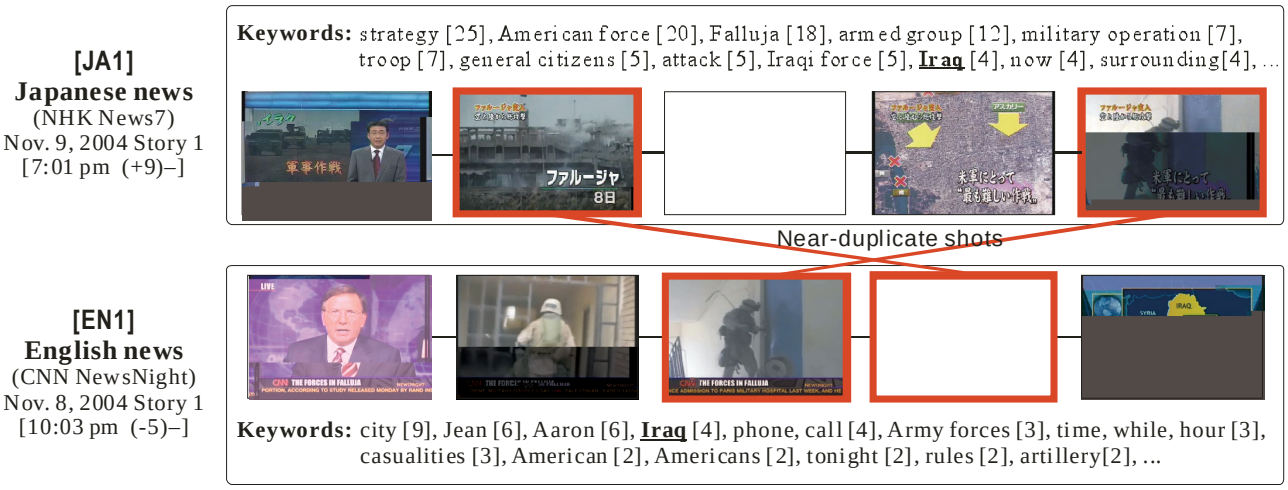


Figure 8: Example of closely related news stories with near-duplicate segments but with few common keywords in the text. Two near-duplicate segments were detected, while only one keyword matched between the two stories. Such a case shows the effectiveness of employing the proposed method.

and timings. Thus, we only compare a fixed region in the vertical center of the frame which is usually not affected by such editing.

For the text comparison, noun compounds and undefined terms (usually proper nouns) are extracted by morphological analysis [4], and the cosine measure between keyword vectors is used to evaluate the similarity, as it was done in 2.2.

In the end, relations obtained from both image and text domains are combined. At this moment, we are still considering how to combine the relations obtained from the two domains.

3.3 Experiment

As a preliminary experiment, news shows listed in Tab. 2 were compared according to the temporal constraints shown in Fig. 7.

As a result, 26 pairs of near-duplicate video segments were found, which covered 4 pairs of news stories discussing the same event, and 1 pair which was actually not a near-duplicate segment. When the stories were compared by text, 4 pairs of news stories discussing the same event were found.

We then compared the pairs, and found that 2 pairs were found by both image and text clues, but 2 pairs were only found by the image clue, which shows the effectiveness of employing the proposed method together with the traditional text-based approach.

Figure 8 shows an example of closely related stories detected by the image clue and not by the text clue.

Table 2: News shows compared in the experiment.

Japanese news [5]	NHK "News7" (Japan: GMT+9) [JA1] Nov. 9, 2004; 7:00 pm í 7:30 pm
English news [6]	CNN "NewsNight" (USA: GMT í5) [EN1] Nov. 8, 2004; 10:00 pm í 11:00 pm [EN2] Nov. 9, 2004; 10:00 pm í 11:00 pm

[4] JUMAN 5.10 distributed from Kyoto University was used.

4 Efficient detection of near-duplicate video segments

This section introduces a general framework that detects all pairs of near-duplicate video segments efficiently from a long video stream (Length: n frames). Near-duplicate video segments are mostly identical video segments from

the image perspective, except for minor local differences such as overlay of captions or logos, or minor overall color difference. They are very important clues to understand semantic structures in video streams. Notice that it is different with the traditional similar video segment detection which detects not only mostly identical but also somewhat similar, but originally different video segments.

Detecting all pairs of near-duplicates in a video stream is, however, extremely time-consuming compared to detecting near-duplicates of a given segment, since it essentially requires computation of a square order of the video length ($O(n^2)$).

Conventional methods approach this problem by accelerating the detection through a two step detection process (Sekimoto et al. 2000, Naturel & Gros 2005, Yamagishi et al. 2003, Yang et al. 2005). The common idea of the methods, including our method, is to make the $O(n^2)$ times comparison as fast and accurate as possible by applying a fast and rough search first and then checking accurately afterwards. These methods do accelerate the detection, but they do not necessarily guarantee that the detection in the first step has no false negatives.

Considering this problem, we propose a method that guarantees that it has no false negatives in the first step, where the results would be theoretically equivalent to the brute-force frame-by-frame comparison. The basic idea of the method is to make the $O(n^2)$ times of comparison as fast and accurate as possible by 1) comparing the features of short video fragments instead of a frame, and also by 2) compressing the dimension of the feature vectors, as shown in Fig. 9. The first approach makes the comparison

[5] Source: NII news video archive
[6] Source: TRECVid 2004 data (US NIST b)