NTT Communication Science Laboratories and National Institute of Informatics at TRECVID 2017 Instance Search

Hidehisa Nagano†, Xiaomeng Wu†, Kaoru Hiramatsu†, Kunio Kashino† and Shin'ichi Satoh‡

† NTT Communication Science Laboratories, NTT Corporation
3-1, Morinosato Wakamiya, Atsugi-shi, Kanagawa, 243-0198 Japan
‡ National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430 Japan
email: nagano.hidehisa@lab.ntt.co.jp

Abstract

We describe our approaches that were tested in the TRECVID 2017 Instance Search (INS) task. In this year's INS, shots including a person at a location were to be retrieved after a location query and person query were given. We used ASMK for the location search and OpenFace for the person search. The results of the location search and person search were fused according to the ranks in the two searches or with the weighted sum of the z-scores of these searches. We also applied human region removal to the sample images for the location search.

We submitted four runs as follows for task A in which no video and only the provided sample images of the queries were used.

NTT_NII_A_run_1 : with human region removal and fusion with rank

NTT_NII_A_run_2 : without human region removal and fusion with rank

NTT_NII_A_run_3 : with human region removal and fusion with weighted sum of z-scores

NTT_NII_A_run_4 : with human region removal and fusion with weighted sum of z-scores

1 Our approach to Instance Search task

The Instance Search task in TRECVID 2017 was to retrieve for each topic up to 1000 shots most likely to contain a query person at a query location, given a collection of test videos, a master shot reference, example images/videos of location queries, and example images/videos of person queries [1].

This year, we used no example video. We used only given example images; i.e., we tried task A only. For each topic, we retrieved shots containing the location given in the topic with the given example images for that location, and retrieved shots containing the person given in the topic with the given example images for this person; then, we integrated these retrieved shots and selected those containing both the location and the person. We tried only fully automatic search.

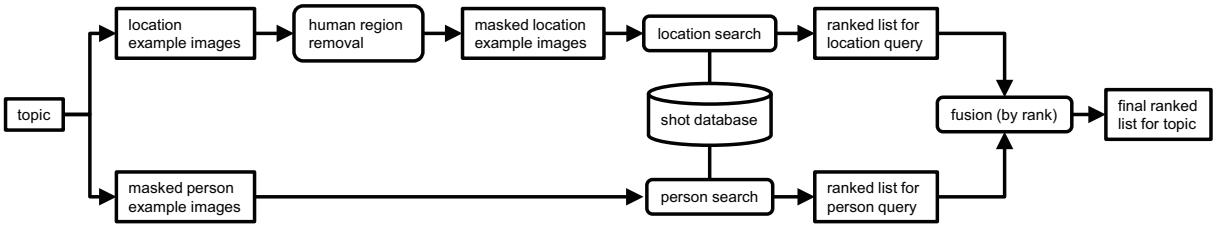

Figure 1: System overview of NTT_NII_A_run_1

2 System Overview

2.1 NTT_NII_A_run_1

Figure 1 shows the flow of the system for NTT_NII_A_run_1. First, we extracted five keyframes per second from the test videos. Then, we constructed a database of shots. Each shot in the database contained the keyframes extracted from it.

Location search: Several example images were given for each query location. First, we used the TensorFlow Object Detection API [2, 3] to remove human regions from the example images. This was done to remove noisy features [4]. Then, we retrieved the shots containing the query location with the Aggregated Selective Match Kernel (ASMK) method [5]. Local features of each keyframe were extracted with a Hessian affine region detector [6] and SIFT descriptor [7]. The local features extracted from all keyframes of a shot were gathered and used to construct a vector representation of the shot. Next, the local features that had been extracted in the same way as above but from all example images for the query location were also gathered and used to construct a vector representation of the query location. Finally, for each query location, the shots were ranked with the scores calculated by the ASMK method.

Person search: We used OpenFace [8, 9] for the person search, and we focused on the person search by using only frontal face images which OpenFace could deal with.

Four masked example images were given for each person query. For each example image and shot, the cosine similarities between the frontal face descriptor extracted from the example image and the frontal face descriptors extracted from keyframes of the shot were calculated with OpenFace; the maximum cosine similarity was regarded as the similarity between the example image and the shot. After that, the similarities between the example image and all shots were normalized as z-scores. Here, for each shot, there were four z-scores for the four example images of a person query, and the maximum score of these four scores was used as the similarity between the shot and the person query. Then, for each person query, the scores between the person query and all shots were normalized as z-scores. These scores were used as the similarities between the person query and the shots.

Fusion (by rank): Given the ranked lists of the person query and location query, the fusion module fuses them and generates the final ranked list of the topic with the person query and location query. Let $r_{location}(shot\_id)$ be the rank of the shot with $shot\_id$ in the location search and $r_{person}(shot\_id)$ be the rank of the shot with $shot\_id$ in the person

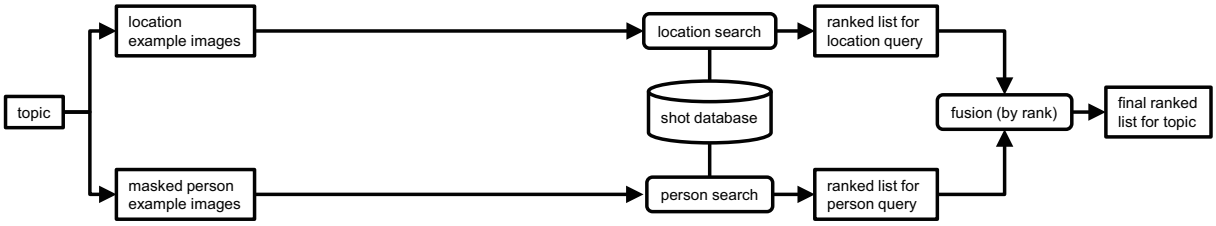

Figure 2: System overview of NTT_NII_A_run_2

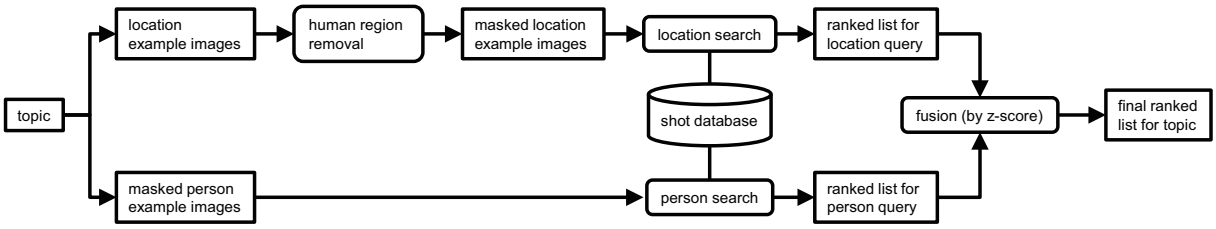

Figure 3: System overview of NTT_NII_A_run_3 and NTT_NII_A_run_4

search; then, the fusion module calculates $\frac{1}{r_{location}(shot\_id)+r_{person}(shot\_id)}$ as the final score of *shot_id* for the topic. By sorting shots with these final scores in descending order, the fusion module generates the final ranked lists of the shots for the topic.

2.2 NTT_NII_A_run_2

Figure 2 shows the flow of the system for NTT_NII_A_run_2. The difference between it and NTT_NII_A_run_1 is the removal of human regions from the example images in only the location search in NTT_NII_A_run_1. The original sample images were used for the location search in NTT_NII_A_run_2.

2.3 NTT_NII_A_run_3 and NTT_NII_A_run_4

Figure 3 shows the flow of the system for NTT_NII_A_run_3 and NTT_NII_A_run_4. The difference between NTT_NII_A_run_3 and _4 is the setting of w_2 in the fusion module. The fusion module fuses the ranked lists as follows.

Fusion (by z-score) Given the ranked lists of the person query and location query, the fusion module fuses them and generates the final ranked list of the topic with the person query and location query. Let $s_{person}(shot\_id)$ be the z-score of *shot_id*, in the ranked list from the person search and $s_{location}(shot\_id)$ be the z-score of the *shot_id* in the ranked list from the location search; then, the fusion module calculates $w_1 \times r_{location}(shot\_id) + w_2 \times r_{person}(shot\_id)$ as the final score of *shot_id* for the topic. By sorting shots with these final scores in descending order, the fusion module generates the final ranked lists of the shots for the topic.

Table 1: Summary of systems

run id	human region removal from sample images in location search	fusion type	MAP
NTT_NII_A_run_1	on	by rank	0.051
NTT_NII_A_run_2	off	by rank	0.037
NTT_NII_A_run_3	on	by z-score ($w_1 = 1.0, w_2 = 20.0$)	0.056
NTT_NII_A_run_4	on	by z-score ($w_1 = 1.0, w_2 = 30.0$)	0.059

Table 2: Number of sample images and those for which face regions were detected and used in person search.

person	#(sample images)	#(used sample images)
Archie	4	1
Billy	4	3
Ian	4	4
Janine	4	2
Peggy	4	2
Phil	4	2
Ryan	4	3
Shirley	4	4
(average)	4	2.6

3 Evaluation of submitted runs

Table 1 shows the evaluation of the submitted runs. The MAP values for all the runs are low. In every run, the person search was performed with OpenFace using only four provided sample images. The non-person regions of these sample images were masked with the provided mask images. In our person search module, first, frontal face regions were selected from each image, and the similarity between the frontal face regions of two images were calculated. In TRECVID 2017, eight persons were given as person queries, and four sample images were given for each person query. However, in our runs, no frontal face region was extracted from some of the sample images. This means that a sample image for which no frontal face region was extracted was not used for the person search. Table 2 shows the numbers of sample images and those for which frontal face regions were detected and used in the person search. For Archie, the frontal face region was detected in only one sample image. The average number of sample images from which frontal face regions were extracted is 2.6, quite a small value. It is conjectured that the frontal face regions extracted from the keyframes were not so many and a satisfactory person search was not performed. We should use not only frontal face images but also images of faces from the side and whole bodies and even views from behind for the person search.

4 Conclusions

We described our approaches and results for the TRECVID 2017 INS task. We used ASMK for the location search and OpenFace for the person search. The results of the location search and person search were fused on the basis of the rank of each search or weighted sum of z-scores. The person search did not work well, so we should improve it.

References

[1] G. Awad, A. Butt, J. Fiscus, D. Joy, A. Delgado, M. Michel, A. F. Smeaton, Y. Graham, W. Kraaij, G. Quenot, M. Eskevich, R. Ordelman, G. J. F. Jones, and B. Huet, "Trecvid 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking," in *Proceedings of TRECVID 2017*. NIST, USA, 2017.

[2] https://research.googleblog.com/2017/06/supercharge-your-computer-vision-models.html.

[3] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[4] D. Le, S. Phan, V. Nguyen, B. Renoust, T. A. Nguyen, V. Hoang, T. D. Ngo, M. Tran, Y. Watanabe, M. Klinkigt, A. Hiroike, D. A. Duong, Y. Miyao, and S. Satoh, "NII-HITACHI-UIT at TRECVID 2016," in *Proceedings of TRECVID 2016*. NIST, USA, 2016.

[5] G. Tolias, Y. Avrithis, and H. Jegou, "To aggregate or not to aggregate: Selective match kernels for image search," in *2013 IEEE International Conference on Computer Vision*, Dec 2013, pp. 1401–1408.

[6] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, Oct 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000027790.02288.f2

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000029664.99615.94

[8] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 815–823.

[9] https://cmusatyalab.github.io/openface/.