THE FAST-3D SPATIO-TEMPORAL INTEREST REGION DETECTOR

Sander Koelstra, Ioannis Patras

Queen Mary, University of London. Mile End Road, London, E1 4NS, UK

{sander.koelstra,ioannis.patras}@elec.qmul.ac.uk

ABSTRACT

Spatio-temporal interest region detectors can be used in the analysis of video to determine sparse, informative regions as candidates for feature extraction. In this paper we compare existing detectors and introduce the new FAST-3D detector, loosely based on the FAST spatial interest region detector. We compare the invariance of detectors to rotation, scale and compression by measuring the similarity between detected interest regions in original and transformed versions of videos. We measure both the repeatibility and introduce a new similarity measure based on mutual information. The FAST-3D detector is shown to be on par with the other detectors, while showing a significant increase in speed.

1. INTRODUCTION

In the area of event detection and recognition from video sequences, an emerging technique is the use of spatio-temporal interest region detection. Locations of interest are those that have a high descriptiveness, can be reliably and repeatedly detected in different instantiations of the same event and are largely invariant to common transformations and noise in the video. Extracting features only at spatio-temporal interest regions allow for a compact, sparse representation of the image sequence, whilst retaining the most significant parts.

A wide range of detectors for static images exists (e.g. SIFT, SURF, FAST, MSER, Salient regions, Harris corners, etc.) and several comparisons are available (i.e.[1, 2]). However, for spatio-temporal interest region detection the field is much smaller, only a few works exist and no comprehensive performance evaluation of the detectors exists. Also, one of the drawbacks of current methods is their computational complexity. Here, we extend the spatial FAST detector into the temporal domain and evaluate it and other detectors in terms of both speed and invariance to common transformations of the video. The FAST-3D detector shows a performance on par with the state of the art, yet on average it performs significantly faster. We compare the FAST-3D detector against four other detectors, which are described briefly below.

The detector of Oikonomopoulos et al. [5] is based on the idea of a saliency metric based on information content. At each pixel, cylindrical neighborhoods are considered, a saliency metric that is defined in terms of the signal (e.g. intensity) entropy is computed for each neighborhood and the maxima that exceed a threshold are selected as interest regions. The authors also use a clustering of the salient regions to reduce the number of regions and gain more stable regions in the process. Both the unclustered and clustered regions are used in our experiments.

The Laptev and Lindeberg [6] detector is based on the spatial Harris interest point operator. For a given video sequence, a linear scale-space representation L is derived by convolution with a separable Gaussian kernel. Then, at each pixel a second-moment matrix M is constructed composed of spatial and temporal derivatives from L averaged with a Gaussian weighting function. Interest points are detected in regions where M has large eigenvalues.

In [7], Dollar et al. propose a detector that is tuned to react to periodic motions as well as spatio-temporal corners. Their interest point operator takes the form $R = (I * g * h_{ev})^2 + (I * G * h_{od})^2$, where g is a spatial Gaussian smoothing kernel and h_{ev} and h_{od} are a temporal quadrature pair of 1D gabor filters. Interest points are defined to lie on the maxima of the given response function.

Cheung and Hamarneh [8] proposed N-SIFT, a temporal extension of the well known 2D SIFT technique. A Gaussian scale-space representation is constructed and a DoG (Difference of Gaussians) image pyramid is derived from it. At each level of the DoG pyramid, the local extrema are found by comparing each voxel of the DoG image against the neighbouring voxels as well as the voxel in the scales above and below (and their neighbouring voxels). A threshold is set on the extrema to get the final set of interest regions.

The rest of the paper is organized as follows: in section 2 the FAST-3D detector is described in detail. Section 3 describes the experiment setup and section 4 reports the results, while section 5 concludes the paper.

2. THE FAST-3D INTEREST REGION DETECTOR

The Feature Accelerated Segment Test (FAST) spatial interest point operator was proposed by Rosten & Drummond [3], is designed for use in a real-time tracking application (it's used in [3] to periodically update the tracked position of a predefined model to the image) and operates on a simple principle.

For each pixel (x, y) in an image with intensity $i_{x,y}$ and given a threshold τ, a circle of 16 surrounding pixels is considered and a corner is detected when 12 contiguous pixels have intensity values either all above $i_{x,y} + \tau$ (a positive corner) or all below $i_{x,y} - \tau$ (a negative corner). The detection can be sped up by initially only testing the 1st, 5th, 9th and 13th pixel since at least three of these pixels must meet the criterion in order for a corner to exist. Thus, pixels failing the test are immediately rejected.

Our FAST-3D spatio-temporal detector is inspired by the FAST detector. Instead of using a circle around each pixel (x, y, t), we consider the set C of the 26 directly neighbouring pixels to (x, y, t) in a 3D space-time neghborhood. As in FAST, we detect a corner when a proportion of the surrounding pixels have intensities that are either all above $i_{x,y,t} + \tau$ (a "positive" corner) or are all below $i_{x,y,t} - \tau$ (a "negative" corner). Requiring that pixels matching the criterion are contiguous does not work well in 3D, since very sharp corners rarely occur. Also, it is much more complex to test whether the pixels are in fact contiguous. Thus, in the 3D-case we detect a corner when a subset C_{sub} of at least 50% of the pixels in C passes the criterion for a negative or positive corner.

The rejection scheme of testing four pixels on the circle in FAST does not translate directly into 3D. We do however propose a simple rejection scheme by requiring a change in both the temporal and the spatial domain. The rejection scheme for a single pixel (x, y, t) in the case of a positive corner is:

1. ($i_{x,y,t-1}$ or $i_{x,y,t+1}$) $> i_{x,y,t} + \tau$, else reject.
2. ($i_{x-1,y,t}$, $i_{x+1,y,t}$, $i_{x,y-1,t}$ or $i_{x,y+1,t}$) $> i_{x,y,t} + \tau$, else reject.
3. (at least 13 pixels in C) $> i_{x,y,t} + \tau$, else reject.

This set of tests is also run for negative corners. While the first test requires a temporal change, the second test demands a spatial change. When a pixel passes all three tests, a spatio-temporal interest region is detected.

In order to apply the detector at scales s we construct a scale-space representation. At each scale s, we set brightness value of a block to be equal to the average brightness level in the 27 adjacent blocks of dimensions $2s_x+1, 2s_y+1, 2s_t+1$. The calculation of these average brightness levels for each pixel and for each scale is computationally quite expensive. Here, we build on the idea of integral images [4] and construct an integral volume V once, from which the average levels can be extracted efficiently for each pixel scale. In this way, we can use the same fast calculation method described earlier for detecting regions of scale 1.

Let $i_{x,y,t}$ denote the intensity values of pixels in the video sequence, then the integral volume is defined as:

$$V_{x,y,t} = \sum_{p=0}^{x-1} \sum_{q=0}^{y-1} \sum_{r=0}^{t-1} i_{p,q,r} \tag{1}$$

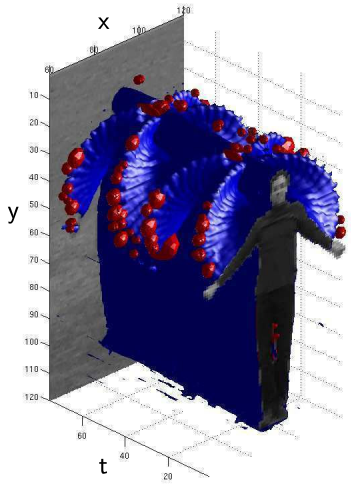

Fig. 1. Example of detected spatio-temporal interest regions. The video shows a person clapping his hands above his head. Interest regions detected by FAST-3D are shown in red.

It can be calculated in one pass over the original volume by using recursion with 8 array references per pixel:

$$\begin{aligned} V_{x+1,y+1,t+1} &= i_{x,y,t} + V_{x,y,t} \\ &\quad - V_{x+1,y,t} - V_{x,y+1,t} - V_{x,y,t+1} \\ &\quad + V_{x,y+1,t+1} + V_{x+1,y,t+1} + V_{x+1,y+1,t+1} \end{aligned} \tag{2}$$

Having V, we can extract a volume A_s for each scale s, with each element $A_{s\,x,y,t}$ the average of the block of dimensions $2s_x + 1, 2s_y + 1, 2s_t + 1$ around pixel (x, y, t). It can be constructed efficiently from I, again using 8 array references per pixel. Let x_+ denote $x + \lfloor s_x/2 \rfloor$, x_- denote $x - \lfloor s_x/2 \rfloor$ and similarly for y and t, then:

$$\begin{aligned} A_{s\,x,y,t} &= s_x^{-1} s_y^{-1} s_t^{-1} (I_{x_+,y_+,t_+} + I_{x_+,y_-,t_-} \\ &\quad + I_{x_-,y_+,t_-} + I_{x_-,y_-,t_+} - I_{x_-,y_-,t_-} \\ &\quad - I_{x_-,y_+,t_+} - I_{x_+,y_-,t_+} - I_{x_+,y_+,t_-}) \end{aligned} \tag{3}$$

If we have n different scales $S = s_1 \ldots s_n$ to detect regions for, this gives a total of $8n+8$ array references to calculate the average brightness level for a pixel and its 26 neighbours at all scales. Without the described technique, the same calculation would take $\sum_{n'=1}^{n} (2n' + 1)^3$ array references.

3. EXPERIMENTS

36 videos from the KTH [9] dataset were used in this experiment, ranging in duration between 9.6 and 29.3 seconds at a resolution of 160x120 pixels. Videos were transformed by zooming or rotating them and regions were detected both in the original and the transformed version. The inverse of the transformation was applied to the regions to get the location and scale corresponding to the regions detected in the

original videos. In addition to these transformations, we also tested the degradation of performance when applying MPEG-compression. For each detector, regions were detected in scales ranging from 3 to 45 pixels across in all dimensions. All detectors were run with parameters set to the authors' defaults. All detected regions were approximated by spheroids to facilitate an easy comparison.

3.1. Performance measures

Two different measures were used to estimate the consistency at which sets of regions are detected in original and transformed videos.

Repeatability as a measure of the performance of spatial interest region detector is proposed in [1]. Two regions a and b with regions R_a and R_b respectively, are deemed a match if the ratio between the union and intersection is large enough, as is detailed in the following equation: Y

$$1 - \frac{R_a \cap R_b}{R_a \cup R_b} < \epsilon \tag{4}$$

Here, ϵ is a threshold which is set at 0.4 by the authors. Then, the repeatibility is defined as the ratio between the number of matches and the smaller of the number of regions detected in each of the two images. In the spatio-temporal case, we take the same approach, calculating the union and intersection numerically by simply counting the number of pixels. However, the repeatability measure suffers from some drawbacks. It favours dense detectors, since as the number of detected regions increases, so does the likelihood that some regions will match by accident. Furthermore, in certain cases a high repeatability value can occur for quite different sets of regions, e.g. when one set of regions is a subset of the other.

To overcome some of these problems, we introduce a measure on an information-theoretical basis as an alternative to the repeatibility measure. For each set of regions, we consider the spatiotemporal coverage by regions of the sequence as a 3D pdf and then consider the mutual information between the two pdfs, that is the amount of information that one of them conveys about the other. First, we estimate a 'coverage matrix' analogous to a probability density function that indicates for each pixel to what degree it is covered by interest regions. This is done by convoluting the center of each interest region with a 3D Gaussian for which $\sigma_x = s_x \cdot 2, \sigma_y = s_y \cdot 2, \sigma_z = s_z \cdot 2$ where s is the detected scale of the region. This gives us a coverage matrix C_a for the original and C_b for the transformed video. Our measure is then a normalised variant of mutual information also known as symmetric uncertainty between these matrices:

$$2 \cdot \frac{H(C_a) + H(C_b) - H(C_a, C_b)}{H(C_a) + H(C_b)} \tag{5}$$

Detector	Avg. FPS	Std	Lang.
FAST-3D	1.6442	0.6002	Matlab
Dollar et al.	1.1000	0.3129	Matlab
Cheung and Hamarneh	0.3734	4.4650	C++
Oikonomopoulos et al. [5]	0.2803	0.6052	Matlab
[5] + clustering	0.2441	0.4943	Matlab
Laptev and Lindeberg	0.0752	0.0331	Matlab

Table 1. Table displaying the average speed of detection over all videos used in the experiment. Shown are the average framerate and its standard deviation.

4. RESULTS

Figure 2 shows the average performance (consistency in region localisation) for zoomed, rotated and compressed videos, respectively. Both in repeatability and mutual information Oikonomopoulos et al. and FAST-3D tend to score the highest. The clustering in the detector of Oikonomopoulos et al., while giving very sparse results, leads to a significant drop in performance. All detectors seem to be reasonably invariant to rotation and zoom, although the repeatability in the case of zooming is quite variable. For MPEG-compression, it seems that except for the FAST-3D and Oikonomopoulos et al. detectors, there is not much invariance to this distortion.

Finally, Table 1 depicts the speeds measured for each of the detectors. The FAST-3D detector appears to run about 50% faster than the second-fastest, (i.e. Dollar et al), and more than 200 times faster than the slowest detector in the test. While the reported execution times depend on the implementation, we note that we have used the implementations provided by the authors (only modifying the scales that are detected for a fair comparison), and all detectors are implemented in Matlab, except for the Cheung and Hamarneh detector, which is in C++.

5. CONCLUSION

In this paper we introduced the FAST-3D detector and showed that it detects spatiotemporal regions consistently when videos are transformed by zoom, rotation or MPEG compresssion. While performing on par with other state-of-the-art detectors, it runs significantly faster. We also introduced a new measure of region consistency based on mutual information that has an information-theoretical grounding. Further work will include testing the detectors in this paper with different descriptors on a realistic dataset to gain insight into the quality of detected regions and a thorough complexity analysis of the detectors.

6. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the Seventh Framework Programme under grant agreement no. FP7-216444.

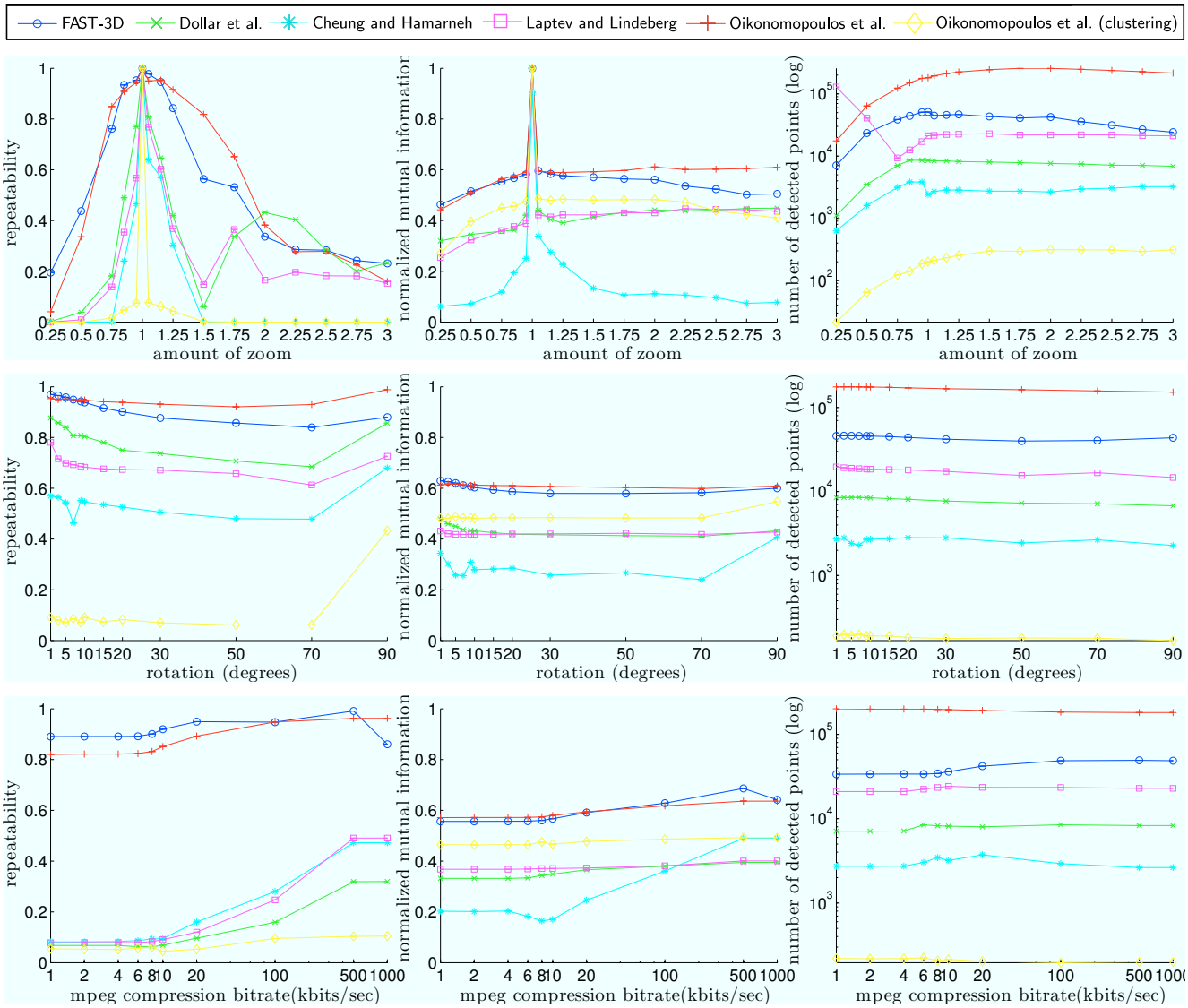

Fig. 2. performance of detectors. The top plot shows the performance as functions of zoom, rotation and compression bitrate.

7. REFERENCES

[1] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L.V. Gool, "A Comparison of Affine Region Detectors," *International Journal of Computer Vision*, vol. 65, no. 1, pp. 43–72, 2005.

[2] N. Sebe and M.S. Lew, "Comparing salient point detectors," *Pattern Recognition Letters*, vol. 24, no. 1-3, pp. 89–96, 2003.

[3] Edward Rosten and Tom Drummond, "Fusing points and lines for high performance tracking.," in *Proc. Int. Conf. Computer Vision*, October 2005, vol. 2, pp. 1508–1511.

[4] P. Viola and M.J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[5] A. Oikonomopoulos, I. Patras, and M. Pantic, "Spatiotemporal salient points for visual recognition of human actions," *IEEE Trans. Systems, Man and Cybernetics*, vol. 36, no. 3, pp. 710–719, 2006.

[6] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. Int. Conf. Computer Vision*, 2003, vol. 1, pp. 432–439.

[7] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *IEEE Int'l Workshop VS-PETS*, 2005, pp. 65–72.

[8] W. Cheung and G. Hamarneh, "n-sift: n-dimensional scale invariant feature transform for matching medical images," in *IEEE Int'l Symposium on Biomedical Imaging*, 2007, pp. 720–723.

[9] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *Proceedings, International Conference on Pattern Recognition*, 2004, vol. 3, pp. 32–36.