02-08

16th International Conference on Machine Vision Applications (MVA)
National Olympics Memorial Youth Center, Tokyo, Japan, May 27-31, 2019.

Temporally Forward Nonlinear Scale Space with Octave Prediction for High Frame Rate and Ultra-Low Delay A-KAZE Matching System

Yuan Li, Songlin Du, Takeshi Ikenaga
Graduate School of Information, Production and Systems, Waseda University,
Kitakyushu 808-0135, Japan
liyuan3104@fuji.waseda.jp

Abstract

High frame rate and ultra-low delay matching system is appealing because of its excellent experience for human-machine interactive applications. A-KAZE algorithm is chosen because of its high robustness and high speed. Nonlinear scale space is very important in A-KAZE, but it not only has at least one frame delay and but also is not hardware friendly. This paper proposes temporally forward nonlinear scale space with octave prediction for high frame rate and ultra-low delay A-KAZE matching system. Problems of complex calculations, data dependency and long time delay are solved by HFD based temporally forward nonlinear scale space with octave prediction. Motion estimation prediction is utilized to improve the robustness. It is also processed parallel with keypoint detection part. It finishes processing before the next frame coming and there is no delay as a results. What's more, lower position gray-coded bit-plane motion estimation has been proposed to improve performance. The results show that the proposed method keeps F-score more than 95% for most cases and shows much better performance compared with the current high frame rate and ultra-low delay matching system.

1 Introduction

Currently, high frame rate and ultra-low delay matching system is aiming at processing speed over 1000 frame per second. It is very attractive because of its excellent experience for many human-machine interactive applications, such as SLAM [1], augmented reality [2] and so on. A more smooth and realistic human-machine interactive experience can be obtained by this system.

Current high frame rate and ultra-low delay matching system [3] reaches 1306 fps with a delay of 0.8083 ms/frame. This system is based on ORB algorithm because ORB uses a hardware friendly keypoint detection algorithm and binary descriptors. It acquires high speed due to its simple algorithm. But at the same time, ORB is not robust enough. To obtain a better performance, the algorithm of matching system need to be improved.

Accelerated KAZE has been proposed in 2011 [4]. It mainly includes three steps, they are nonlinear scale space generation, keypoint detection and descriptor generation separately. Because A-KAZE algorithm's nonlinear scale space adopts nonlinear diffusion [5] to generate scale space, it keeps more edge details compared with SIFT [6] algorithm's scale space. A-KAZE algorithm shows a similar or even better performance compared with SIFT. At the same time, the descriptor of A-KAZE is binary descriptor, A-KAZE reaches much higher speed compared with SIFT.

From the above, it is easily found that nonlinear scale space is one of the most important parts of A-KAZE algorithm. But this part not only has a long time delay but also is not hardware friendly. Nonlinear scale space does complex iterations many times and each sublevel [7] needs to wait the results of previous sublevel except the initial sublevel. There is at least one frame delay as a result. It does not meet the requirements of high frame rate and ultra-low delay matching system. What's more, many difficulties of hardware implement are also existed. Firstly, for each sublevel, nonlinear diffusion needs to be implemented. It is widely known that in the nonlinear diffusion equation, there are derivatives and divisions which are all not hardware friendly. Secondly, A-KAZE algorithm adopts unfixed number of iterations to approximate the results of nonlinear diffusion equation step by step, unfixed number of iterations is not able to implement in hardware. What's more, the following sublevel uses the previous sublevel's results to do nonlinear diffusion, data dependency between octaves also exists. To conclude, long time delay, complex calculations, unfixed number of iterations and data dependency exist, nonlinear scale space is difficult to implement in high frame rate and ultra-low delay matching system.

To remove complex calculations and unfixed number of iterations, this paper adopts HFD algorithm [8] to replace nonlinear diffusion equation. To realize high frame rate and ultra-low delay and deal with data dependency, temporally forward nonlinear scale space which uses octave prediction is proposed. Lower position gray-coded bit-plane based motion estimation prediction is proposed to improve the performance. What's more, these processes are parallel with keypoint detection and descriptor generation parts, there is no delay as a result. The whole proposed nonlinear scale space finishes in one frame.

2 Proposed nonlinear scale space

Figure 1 shows the concept of proposed method. The main difference of concepts between proposed method and A-KAZE is that one octave is forward in proposed method. And through parallelism, proposed method is twice faster than the orignal A-KAZE.

2.1 Structure of proposed nonlinear scale space

The comparisons between structures of original A-KAZE algorithm and proposed method are shown in

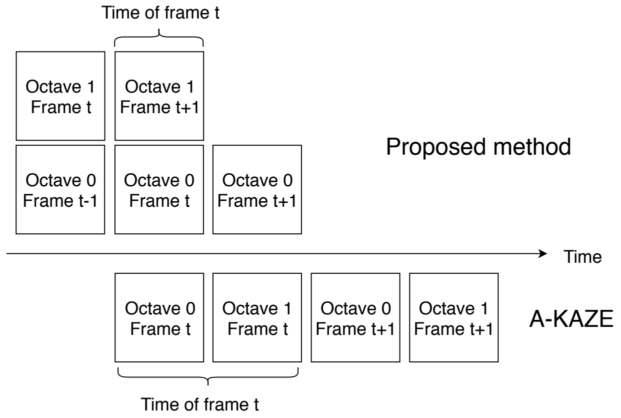


Figure 1: Concept of differences between A-KAZE and proposed method.

Figure 2. Assume that nonlinear scale space only contains 2 octaves and each octave contains 2 sublevels. Figure 2(a) shows the structure of original A-KAZE. Data of frame t is used to do nonlinear diffusion to generate sublevel 0, results of sublevel 0 are used to do nonlinear diffusion to obtain sublevel 1. Octave 1 is similar with octave 0, the only difference is that the initial image of octave 1 is the downsampled sublevel 1 of octave 0. Moreover, the whole nonlinear scale space needs to be built before keypoint detection part. There is a long delay as a result. Structure of proposed method is shown in Figure 2(b). High frame rate image video is used as the input of this algorithm. Octave 0 is generated by using frame t data and using HFD algorithm to do nonlinear diffusion. Octave 1 is generated by downsampled octave 0 of frame t-1. Motion estimation which is calculated by frame t-1 and frame t-2 is used to predict the motion between frame t-1 and frame t. Adding this motion estimation to octave 1 is aimed to improve the performance. The whole operation of octave 1 is finished before the process of frame t. In addition, the calculation process of motion estimation and downsampling are parallel with keypoint detection and descriptor generation part. It implements ultra-low delay for high frame rate video.

2.2 Motion estimation prediction with HFD based temporally forward nonlinear scale space

Hardware friendly descreening (HFD) [8] is widely used for descreening solution. The function of it is similar with nonlinear diffusion filter. HFD algorithm extracts a spatial feature vector comprising the intensity gradients computed at different pixel locations in a small neighborhood of the given pixel. HFD just utilizes multiplication, addition and bit-wise shift, it also does not need to do many unfixed times of iterations. It is hardware friendly. If HFD is used as nonlinear diffusion, nonlinear scale space can be implemented in hardware, but there still is data dependency between octaves. Also, it is not possible to process high frame rate image video with low delay.

To deal with data dependency between octaves and a long delay, temporally forward nonlinear scale space is proposed. It takes advantage of the high temporal coherence of high frame rate images. The octave 0's results of previous frame are utilized. Furthermore, the whole octave 1 of current frame is generated in the process of the previous frame. This octave is predicted and delay will also be decreased as a result. But using previous frame's data makes the robustness decrease, because there are many differences between current frame and previous frame. To deal with this problem, motion estimation prediction is proposed.

Motion estimation [9] is widely used in video coding. From many motion estimation algorithms, selective gray-coded bit-plane based low-complexity motion estimation [10] stands out because of its high processing speed and good performance. Gray code is chosen because it is more robust than binary as successive gray code words differ in only one bit position. It firstly generates the three highest position gray-coded bit-plane of the input image because these three bit-planes include most significant information of input image. And then through EXOR calculations to find the most similar motion to do motion estimation. Motion estimation of frame t-2 and frame t-1 is used to predict the motion from frame t-1 to frame t. However, selective gray-coded bit-plane based low-complexity motion estimation is not very suitable for the target, lower position gray-coded bit-plane based motion estimation is proposed. It will be explained in detail in the next section.

In conclusion, proposed nonlinear scale space is that this frame's nonlinear scale space just uses HFD algorithm to generate octave 0 of this frame. The process of generating just one octave by HFD can be completed before the next frame coming. And the process of using previous frame and current frame's data to calculate motion estimation to do prediction is parallel with keypoint detection part. Adding motion estimation to octave 0 and downsampling is also parallel with descriptor generation part. So, the whole octave 1 of the next frame can be generated before the next frame coming, it is temporally forward. High frame rate and ultra-low delay A-KAZE matching system is to be implemented by means of this proposal.

2.3 Lower position gray-coded bit-plane based motion estimation

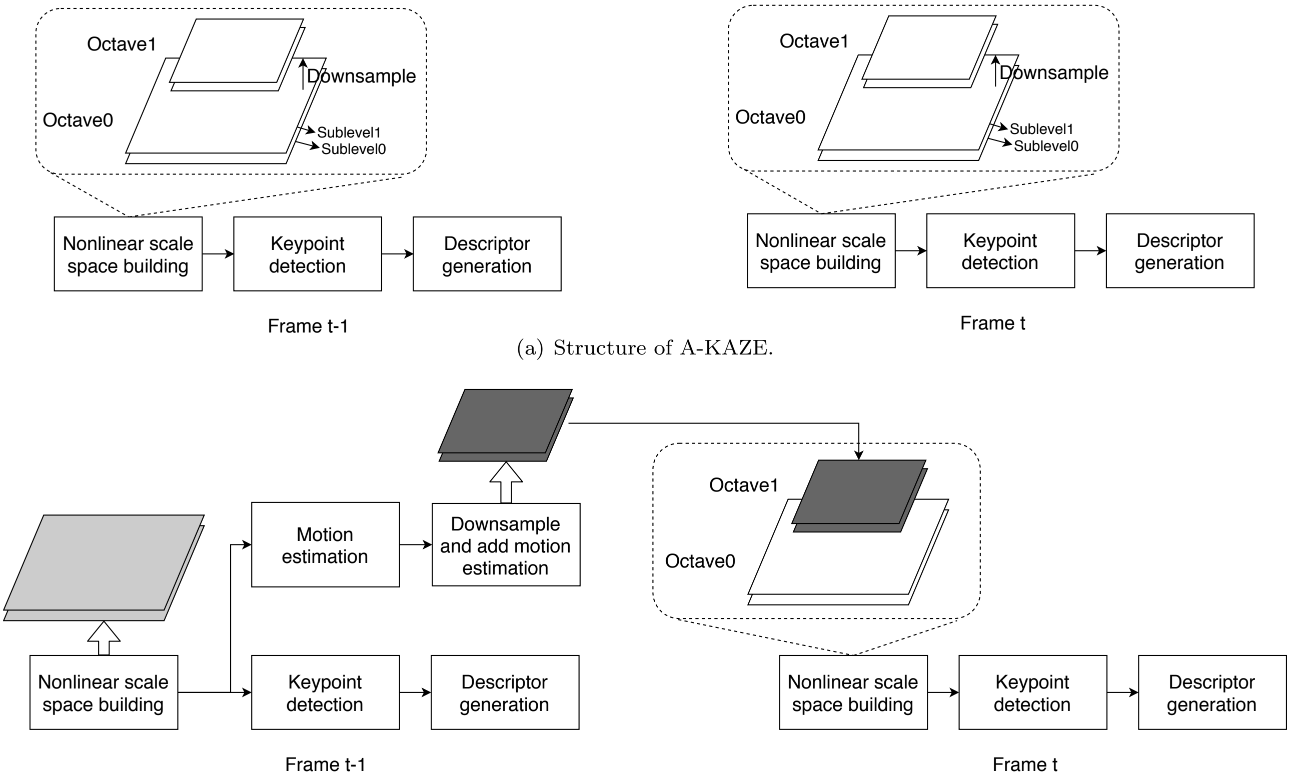
The K-bit gray code of a pixel value is computed by

$$g_{K-1} = a_{K-1}, \tag{1}$$

$$g_k = a_k \oplus a_{k+1}, 0 \leq k \leq K-2, \tag{2}$$

where a means the binary code of a pixel value, g means the gray code of this pixel value. The higher k is, the more important the binary code is. Because higher position binary code influences the pixel value more than lower one. At the same time, it will be less detailed. For example, if a_7 changes from 1 to 0, the pixel value will be 125 lower. But if a_0 changes from 1 to 0, the pixel value will only be 1 lower. The importance of k in gray code is similar with the binary code.

Selective gray-coded bit-plane based low-complexity motion estimation chooses the three highest position bit-planes, because they are considered to be the three most important bit-planes [11]. But they are not the most suitable for motion estimation prediction because the motion information is the most needed. Through



(a) Structure of A-KAZE.

(b) Structure of proposed method.

Figure 2: Comparisons of structure between A-KAZE and proposed method.

the analysis of gray coded bit-planes, it is found that in g_7, although it includes the most important information, it does not include any motion information. It just has a large area of white or black. What's more, the lowest four bit-planes include too many detailed information, the background and other not needed information will influence the object too much to obtain accurate motion information. As a result, g_6, g_5 and g_4 these three lower position gray coded bit-planes which obtain accurate motion information are chosen to do the motion estimation prediction.

3 Evaluation results

Test sequences are all captured by high frame rate camera, they include 1200 frames for each sequence. What's more, the resolution of each image is 640 × 480. Parallel translation, in-plane rotation, change of illumination and scale change, these four most representative situations are included in the test sequences. Parallel translation means the object just moves from top to bottom back and forth. In the situation of rotation, the object is in the center of the images and rotates in the plane. Illumination change means that the object does not move and just changes the brightness of the surroundings. Lastly, for scale change, just the size of the object is changed, the object moves from far side of the camera to near side. Figure 3 shows the four sequences which are used to test the performance of algorithms.

Comparisons of F-score among five algorithms are shown in Table 1. The first one with motion estimation prediction means the whole proposed method. The second one is without motion estimation prediction, it just uses HFD algorithm to replace nonlinear diffusion and just downsamples the octave 0 of previous frame to be the octove 1 of this frame and does not do motion estimation. Without motion estimation prediction means just temporally forward nonlinear scale space. The third one is HFD which means just uses HFD algorithm to replace the nonlinear diffusion part of A-KAZE. What's more, the forth one and the fifth one are original A-KAZE algorithm and current high frame rate and ultra-low delay ORB matching system, respectively. The data are all percentage form. Through the analysis on results of F-score, it is absolutely that A-KAZE matching system shows much higher performance compared with ORB algorithm based matching

Table 1: F-score (%) of five algorithms.

	With motion estimation prediction	Without motion estimation prediction	HFD	A-KAZE	ORB matching system
Translation	95.57	93.53	96.19	96.23	89.96
Rotation	91.25	91.51	96.39	97.20	86.94
Illumination change	96.61	95.74	97.60	98.68	92.30
Scale change	96.82	94.22	97.45	97.43	89.60

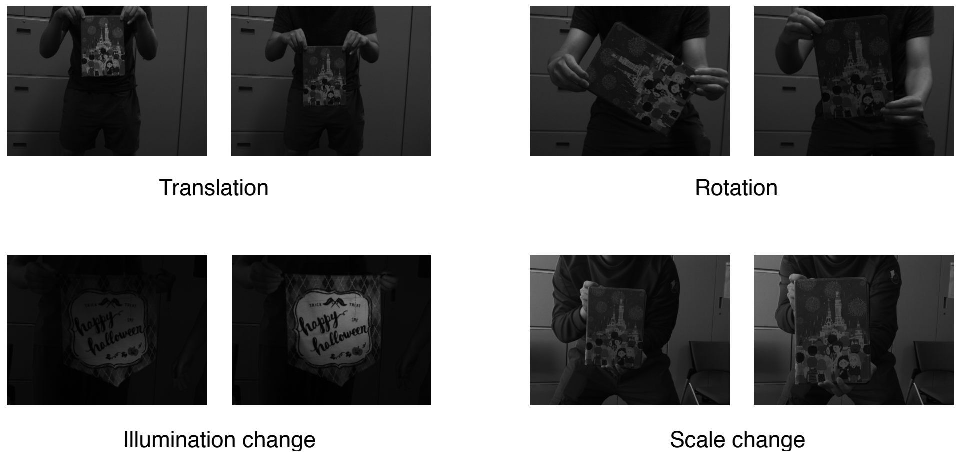


Figure 3: Sequences used for test performance.

system. A-KAZE matching system makes up for the low robustness of ORB matching system. Using HFD algorithm to replace nonlinear diffusion still keeps high performance. As a result, it is reasonable to use HFD to replace nonlinear diffusion filter. Simple temporally forward nonlinear scale space which does not do motion estimation prediction results in just 2 percent lower F-score in the cases of translation, illumination change and scale change. However, it shows a bad performance for rotation, there is 4.88 percent decrease of F-score. Lower position gray coded bit-plane motion estimation prediction is added to improve the performance. The results show that after adding motion estimation prediction, F-score improves more than 1 percent for the situations of translation, illumination changes and scale changes, and it is close to the performance of HFD. But for rotation, its performance is still not as good as the algorithm without motion estimation prediction.

Through analysis, the reason of low robustness on rotation is that motion of rotation changes hugely from frame to frame compared with the other situations. Because motion for each frame is different, the differences between adding previous frame's motion vector to current frame are much more than the differences between the previous frame and current frame, while the aim is to make these two more similar to improve matching performance.

4 Conclusion and future work

This paper proposes motion estimation prediction with temporally forward nonlinear scale space for high frame rate and ultra-low delay A-KAZE matching system. HFD algorithm is utilized to replace nonlinear diffusion, the downsampled previous frame's octave 0 is used to act as the octave 1 of current frame because of the high temporal coherence of high frame rate images. What's more, lower position gray-coded bit-plane motion estimation prediction is added to improve matching performance. This process is parallelizable, so there is no delay for implementation of high frame rate and ultra-low delay matching system. The results show that the proposed method keeps high performance in most cases and much better than current ORB matching system. In future work, robustness for rotation needs to be improved and hardware resources need to be taken into consideration.

References

[1] Dissanayake, MWM Gamini, et al.: "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on robotics and automation*, vol.17, no.3, pp.229-241, 2001.

[2] AZUMA, Ronald, T.: "A survey of augmented reality," *Presence: Teleoperators and Virtual Environments*, vol.6, no.4, pp.355-385, 1997.

[3] Hu, T., Ikenage, T.: "Pixel Selection and Intensity Directed Symmetry for High Frame Rate and Ultra-Low Delay Matching System," *IEICE TRANSACTIONS on Information and Systems*, vol.E101-D, no.5, pp.1260-1269, 2018.

[4] Alcantarilla, Pablo F., Solutions, T.: "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE Trans. Patt. Anal. Mach. Intell*, vol.34, no.7, pp.1281-1298, 2011.

[5] Perona, P., Malik, J.: "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on pattern analysis and machine intelligence*, vol.12, no.7, pp.629-639, 1990.

[6] Lowe, David G.: "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol.60, no.2, pp.91-110, 2004.

[7] Vedaldi, A.: "An implementation of SIFT detector and descriptor," *University of California at Los Angeles*, vol.7, 2006.

[8] Siddiqui, H., Boutin, M., et al.: " Hardware-friendly descreening," *IEEE Transactions on Image Processing*, vol.19, no.3, pp.746-757, 2010.

[9] Rao, K. R., Hwang, J. J. : " Techniques and standards for image, video, and audio coding," *New Jersey: Prentice Hall*, vol.70, 1996.

[10] Yavuz, S., Celebi, A., et al.: " Selective gray-coded bit-plane based low-complexity motion estimation and its hardware architecture," *IEEE Transactions on Consumer Electronics*, vol.62, no.1, pp.76-84, 2016.

[11] Natarajan, B., Bhaskaran, V., et al.: " Low-complexity block-based motion estimation via one-bit transforms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.7, no.4, pp.702-706, 1997.