

Feature matching algorithm based on KAZE and fast approximate nearest neighbor search

Ze-Ping Cai

College of Information Science and Engineering
Hunan University
Changsha, China
caizeping2009@163.com

De-Gui Xiao

College of Information Science and Engineering
Hunan University
Changsha, China
dgxiao@hnu.edu.cn

Abstract—This paper proposed a feature matching algorithm based on KAZE and fast approximate nearest neighbor search for that SIFT and SURF feature detection algorithm, extracting features by Gaussian pyramid in the linear scale space has the problems of fuzzy boundaries, detail missing, and low feature points matching rate. First, this algorithm uses Additive Operator Splitting(AOS) method for nonlinear diffusion filtering, and structure nonlinear scale space. Then use Hessian matrix to detect feature points, and construct the Gauge-SURF(G-SURF) descriptions in the Gauge coordinate. Finally adopt fast approximate nearest neighbor search algorithm to match feature points, and use RANSAC algorithm to eliminate false matching points. Experiments show that this algorithm ensures the real-time nature and improves the feature matching rate.

Key words—Feature match; Nonlinear scale space; KAZE; The Gauge-SURF; FLANN

I. INTRODUCTION

Feature matching has been one of the focus in the research of the image processing. It is extensive applied in video stitching, 3D reconstruction, object recognition and detection, and other fields. The mage matching algorithm based on the feature can overcome the problem of large amount of calculation and can solve the problem of image rotation and scale change better, relative to the image matching algorithm based on gray and the matching algorithm based on region.

The previous algorithms extract feature with the gaussian pyramid structure in the linear scale space generally. For example, Lowe's SIFT (scale invariant feature transform) feature detection algorithm [1]. A feature descriptor based on scale space put forward in 2004 has good invariance and good ability to match in image translation, rotation, affine transformation. And Bay's SURF algorithm [2] proposed in 2006, which is modified by SIFT algorithm is faster than SIFT several times, and has better robustness in multiple images. SURF accelerated the speed of operation greatly with Harris features and integral image. But the gaussian decomposition of SIFT and SURF feature detection algorithm lost local precision, and it is easy to cause boundaries and details loss. And features matching rate are not very high using feature descriptions such as SURF, M-

SURF.

This paper proposed a feature matching algorithm based on KAZE [3] and fast approximate nearest neighbor search[4] for the advantages and disadvantages above mentioned. First constructed nonlinear scale space using AOS algorithm, and detected features using the Hessian matrix [5]. Then generated 64 dimensional feature vector using Gauge-SURF feature descriptor[6]. Finally adopted FLANN (fast approximate nearest neighbor search) algorithm for feature matching, and used RANSAC algorithm for precise matching.

II. KAZE FEATURE EXTRACTION AND DESCRIPTION

This paper adopts KAZE algorithm as the feature extraction algorithm for feature matching, and the extracted features are described by using 64 dimensional Gauge-SURF feature descriptor vector.

A. KAZE feature point extraction

KAZE feature employs the normalized Hessian matrix to detect local maximum and determines the main direction of the feature points in nonlinear scale space.

Nonlinear scale space is constructed with the method of AOS algorithm and variable conduction diffusion. Nonlinear diffusion filtering model [7] describes the evolution of the luminance of an image in different scales as the divergence of a certain flow function:

$$\frac{\partial L}{\partial t} = \text{div}(c(x, y, t) \cdot \nabla t) \quad (1)$$

where L is the luminance of an image, c(x,y,t) is the flow function, and

$$c(x, y, t) = g(|\nabla L_{\sigma}(x, y, t)|) \quad (2)$$

∇L_{σ} is the gradient of a Gaussian smoothed image L_{σ} of the original image[10], and

$$g = \frac{1}{1 + \frac{|\nabla L_\delta|^2}{k^2}} \quad (3)$$

Where K is the contrast factor that controls the level of diffusion, and K is related to to keep the edge information, the greater its value is, the less it can keep edge information. We think of K as the 70% percentile of the gradient histogram ∇L_δ in this paper.

The discretization of equation (1) can be expressed as

$$\frac{L^{i+1} - L^i}{\tau} = \sum_{l=1}^m A_l(L^i) L^{i+1} \quad (4)$$

the solution of this equation is effective to arbitrary time step.

Where A_l is a conductivity matrix[8] on each dimension l of the image. KAZE algorithm uses O octaves and S sub-levels like SIFT respectively. Here we work with the original image resolution at each octave unlike downsampling in SIFT. and

$$\delta_i(o, s) = \delta_0 2^{o + \frac{s}{S}} \quad (5)$$

where $i \in [0, \dots, N]$, $o \in [0, \dots, O-1]$, $s \in [0, \dots, S-1]$,

δ_0 is the base scale parameter.

$N = O * S$ refers to the images of the whole scale space. Then mapping the scale parameter in pixels to the nonlinear

scale space in time unit. t_i is the evolution time.

$$t_i = \frac{1}{2} \delta_i^2 \quad (6)$$

We can get all the images in nonlinear scale space using AOS algorithm according to a set of evolutionary time,

$$L^{i+1} = (I - (t_{i+1} - t_i) \sum_{l=1}^m A_l(L^i))^{-1} L^i \quad (7)$$

Similar to SIFT, KAZE extracts feature points with Hessian matrix [5], Hessian is calculated as follows:

$$L_{Hessian} = \delta^2 (L_{xx} L_{yy} - L_{xy}^2) \quad (8)$$

Where L_{xx} , L_{yy} , L_{xy} representative the second order derivatives, δ is the integer value of scale parameter δ_i . Then judging Hessian extremum points by comparing the size of point between the detecting point and 26 points around it, in the $3 \times 3 \times 3$ neighborhood between three 3×3 rectangular window of the current scale and

adjacent scales. If the point has the extreme value, it is the feature point.

B. Gauge - SURF feature vector describe

G-SURF can not only keep the edge information, but also improve the matching rate of feature points relative to SIFT and SURF descriptors. Then calculate the first-order and second-order haar response L_x , L_y , L_{yy} , L_{xy} in the size of 20×20 (s is the scale factor) around the feature points. L denotes the convolution of the image I with a 2D Gaussian kernel $g(x, y, \sigma)$, where σ is the kernel's scale parameter:

$$L(x, y, \delta) = I(x, y) * g(x, y, \delta) \quad (9)$$

\vec{w} is the gradient vector of feature points and \vec{v} is the gradient's vertical vector in the gauge coordinates (\vec{w}, \vec{v}) , as follows:

$$\vec{w} = \left(\frac{\partial L}{\partial x}, \frac{\partial L}{\partial y} \right) = \frac{1}{\sqrt{L_x^2 + L_y^2}} \cdot (L_x, L_y) \quad (10)$$

$$\vec{v} = \left(\frac{\partial L}{\partial y}, -\frac{\partial L}{\partial x} \right) = \frac{1}{\sqrt{L_x^2 + L_y^2}} \cdot (L_y, -L_x) \quad (11)$$

Then calculate the second order Gauge derivative L_{ww} , L_{vv} ,

$$L_{ww} = \frac{L_x^2 L_{xx} + 2 \cdot L_x L_y L_{xy} + L_y^2 L_{yy}}{L_x^2 + L_y^2} = \frac{1}{L_x^2 + L_y^2} (L_x, L_y) \begin{pmatrix} L_{xx} & L_{xy} \\ L_{yx} & L_{yy} \end{pmatrix} \begin{pmatrix} L_x \\ L_y \end{pmatrix} \quad (12)$$

$$L_{vv} = \frac{L_y^2 L_{xx} - 2 \cdot L_x L_y L_{xy} + L_x^2 L_{yy}}{L_x^2 + L_y^2} = \frac{1}{L_x^2 + L_y^2} (L_y, -L_x) \begin{pmatrix} L_{xx} & L_{xy} \\ L_{yx} & L_{yy} \end{pmatrix} \begin{pmatrix} L_y \\ -L_x \end{pmatrix} \quad (13)$$

We take 25 regularly distributed sample points in each subregion yields, and then calculate a four-dimensional descriptor vector $d_v = (\sum L_{ww}, \sum L_{vv}, \sum |L_{ww}|, \sum |L_{vv}|)$, and generate a 64 - dimensional descriptor vector[11].

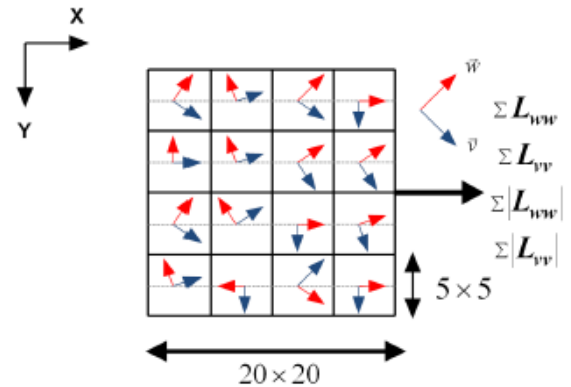


Figure 1 .Gauge - SURF descriptor vector structure

III. FEATURE MATCH

It's time consuming use the nearest neighbor search algorithms such as k - d tree for high-dimensional vector space. In order to reduce the matching time and increase the feature points matching rate, we use FLANN(fast approximate nearest neighbor search) algorithm, which can choose different precision and select different parameters quickly. First, FLANN find the matching point pair which has the recent Euclidean distance d , then eliminate the false matching points through RANASC.

$$d = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + \dots + (x_{64} - x_{64}')^2} \quad (14)$$

Where $(x_1, x_2, \dots, x_{64})$ and $(x_1', x_2', \dots, x_{64}')$ represent the 64-dimensional G-SURF descriptor vector for matching feature points pair respectively.

We eliminate false matching feature points [9] using RANASC algorithm in order to improve the matching efficiency, for it exists some false matching inevitably using FLANN algorithm.

IV. EXPERIMENT RESULT AND ANALYSIS

In order to verify the performance of this algorithm, we implement the algorithm of this paper in VS2010 using C. We find that the proposed algorithm has improved in run time and feature points matching rate through the matching experiment with random images. Figure 2 are two cars pictures which has some overlap. Figure 3 are the feature points with KAZE to extract and with G-SURF to describe. (a) detects 709 feature points , and (b) detects 805 feature points. Figure 4 is the matching use this paper's algorithm, the effective matching points are 34, and the correct matching rate reaches 85%.

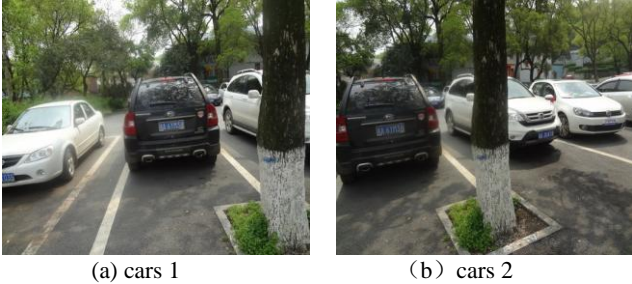


Figure 2. Original experimental images

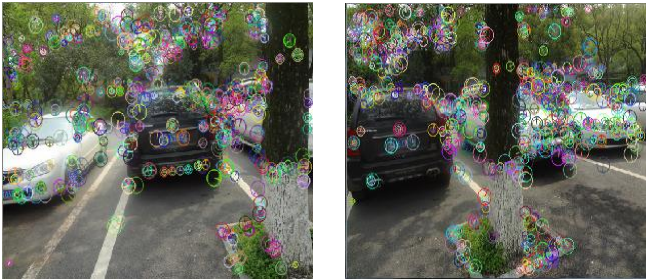


Figure 3. Feature points using KAZE to extract

TABLE I. THE COMPARE OF MATCHING EFFICIENCY BETWEEN

	matching point pairs	Matching rata
this paper's algorithm	34	85.00%
SURF+FLANN	51	79.68%
G-SURF+KNN	115	62.16%



Figure 4. Matching use this paper's algorithm

From table I , feature points matching rata has been improved relative to using single SURF extracting feature because this algorithm uses KAZE to extract features and use G-SURF to describe features. The matching accuracy also will has different with using the same feature extraction and description method but different methods to match. From table 1, we can see matching accuracy has been increased greatly using FLANN algorithm relative to KNN algorithm.

Figure 5 is the matching use SIFT to extract features and use FLANN to match the feature points for the image in figure 2. Table II is the comparison of feature matching between SIFT and KAZE. The matching rate use KAZE feature have a certain degree of improvement.



Figure 5. Matching use SIFT feature points

TABLE II. COMPARISON OF FEATURES MATCHING BETWEEN SIFT AND KAZE

	Number of Matches	Number of Inlier	Match rate
SIFT	11	9	81.82%
KAZE	45	40	88.90%

V. CONCLUSION

This paper researched the feature matching algorithm based on KAZE and fast approximate nearest neighbor search. First, it use KAZE to extracted the feature points in nonlinear scale space, then described the extracted feature points with Gauge-SURF vector. Finally matched feature points in the method of fast approximate nearest neighbor search, and use RANSAC to eliminate false matching points. Experiments show that matching efficiency has been promoted comparing with previous methods such as SURF. Next step we will study in terms of the speed of the algorithm, in order to achieve real-time requirements of this algorithm, and study how to apply the algorithm in image stitching.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (No. 61272062).

REFERENCES

- [1] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. J. of Computer Vision* 60 (2004) 91–110
- [2] Bay, H., Ess, A., Tuytelaars, T., Gool, L.V., "SURF: Speeded up robust features," *Computer Vision and Image Understanding* 110 (2008) 346–359
- [3] Pablo F. Alcantarilla, Adrien Bartoli, Andrew J. Davison, "KAZE Features," *Eur. Conf. on Computer Vision (ECCV)*, 2012
- [4] Marius Muja, David G. Lowe, "Fast Approximate Nearest Neighbors With Automatic Algorithm Configuration," *VISAPP*, 2009
- [5] Lindeberg, T., "Feature detection with automatic scale selection," *Intl. J. of Computer Vision* 30 (1998) 77–116
- [6] Pablo F. Alcantarilla, Luis M. Bergasa, Andrew J. Davison, "Gauge-SURF Descriptors," *Image and Vision Computing*, 2013
- [7] Perona, P., Malik, J., "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.* 12 (1990) 1651–1686
- [8] Weickert, J., ter Haar Romeny, B., Viergever, M.A. "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Processing* 7 (1998)
- [9] Zhao lulu, Geng Guohua, Li Kang, He A-Jing, "Image matching algorithm based on SURF and fast approximate nearest neighbor search," *Computer application research*, 2012
- [10] Weickert, J., "Efficient image segmentation using partial differential equations and morphology," *Pattern Recognition* 34 (2001) 1813–1824
- [11] J. Damon, "Local Morse theory for solutions to the heat equation and Gaussian blurring," *Journal of Differential Equations* 115 (2) (1995) 368–401.