Feature matching using ultrasound images

shuangchao wang(shuangch@ualberta.ca)
yanquan chen(yanquan@ualberta.ca)
xudong li(xudong9@ualberta.ca)

October 8, 2020

1 Abstract

A feature matching method of ultrasonic image should be designed. According to the existing feature matching algorithm, we intend to focus on the direction of improving the traditional algorithm rather than adopting the deep learning method. We plan to put the improvement direction in the algorithm part of feature detection to realize more effective feature detection of ultrasonic images. Thus, it can assist users to match ultrasonic images quickly and accurately, and improve the matching efficiency of ultrasonic image features.

2 Introduction

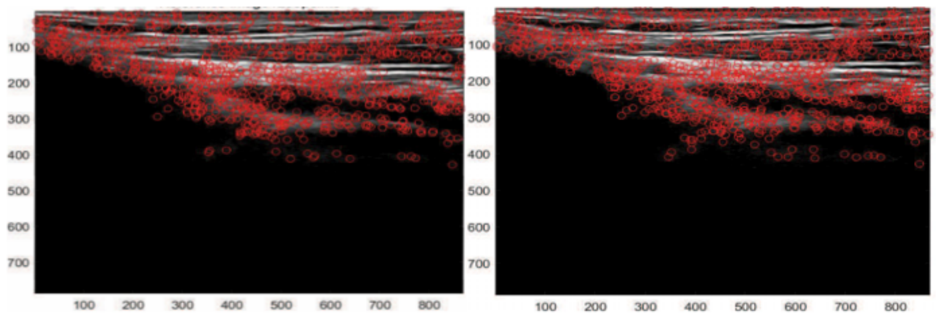


Figure 1: feature detecting and matching between two ultrasonic images

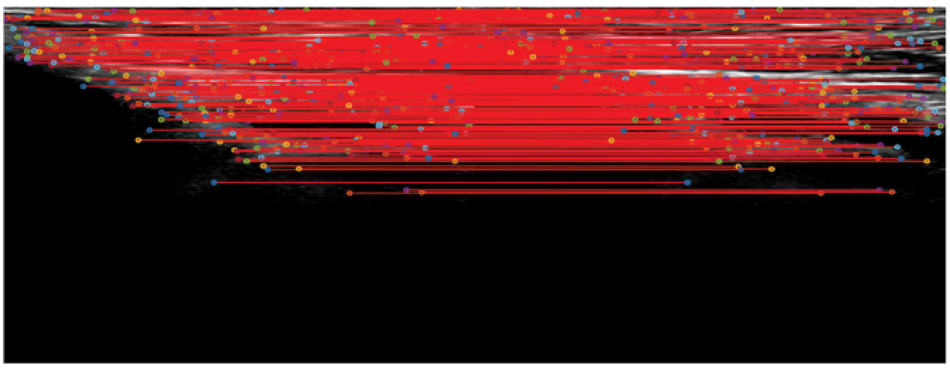


Figure 2: FAST feature detection pixel number and detection range

With the increasing level of science and technology, the development of artificial intelligence technology is more and more rapid, and its main development requirement is to let computers like human beings in feeling, understanding and action. As one of the main perception technologies, computer vision plays a very important role in the field of artificial intelligence. Therefore, it has become an important project in the field of computer vision to understand the relationship between multiple visual objects and process them according to the requirements. Feature matching is one of the key processes. In traditional image processing, image feature matching has three basic steps: feature detection, feature extraction and feature matching. Firstly, feature detection is the significant structural features with physical significance detected from the image, including feature points, feature lines or edges and significant morphological areas. After that, feature extraction will be processed : we need to extract the Location, Orientation and Scale information of the feature. Direction and scale information is primarily intended to support rotation and scale variation. Finally, for feature matching, it is necessary to use certain methods to further determine whether the corresponding features are the same (or approximate). Generally, The eigenvectors are judged by the Use of European distance. If certain constraints are met, the two features will be considered to be similar, or they will be eliminated. Therefore, in traditional image processing, designing a high-precision and high-efficiency matching method has been a hot topic in the past, such as SIFT(Scale-invariant feature transform), FAST(Features From Accelerated Segment Test), SURF(Speeded Up Robust Features), BRIEF(Binary Robust Independent Elementary Features)and other algorithms. They followed the traditional idea based on image matching to improve the extraction and matching ability of feature points to a certain extent, such as extract more accurate and repeatable features, more prominent and distinguishable feature descriptors. The other is to depart from the traditional matching method and adopt the method based on deep learning to match the images, such as Learned Invariant Feature Transform, Discriminative learning of deep convolutional Feature point descriptors, MatchNet, Universal Correspondence Network, etc. At present, deep learning method, because of its superior learning and expression ability for deep features, is applied in various fields of computer vision, and it also stands out in the problem of image matching and has achieved preliminary results.

3 Brief Summary of Existing Work

In this section, we cover the progression of image features detection and matching algorithms over three sections. In the first section, traditional feature detection algorithms will be introduced. After that, we will discuss feature detection based on some deep learning algorithms and techniques. Finally, features matching algorithms will be described in the third section.

3.1 Traditional Algorithms

Based on the existing researches, some traditional computer vision techniques for feature detection includes the following: Harris Corner Detection, Shi-Tomasi Corner Detector, Scale-Invariant Feature Transform(SIFT), Speed-Up Robust Features(SURF), Features from Accelerated Segment Test(FAST), Binary Robust independent Elementary Features(BRIEF), Oriented FAST and Rotated BRIEF(ORB). In 1988, Chris Harris and Mike Stephens introduced the Harris Corner Detection algorithm to find corners in the image with large variation in intensity in all directions, it applies a Gaussian window on the image and determines whether a window contains a corner. It was also one of the earliest methods for corner feature detection [8]. In 1994, J. Shi and C. Tomasi made a small modification on the scoring function of the original Harris Corner Detection algorithm and showed better results. This improved algorithm is called Shi-Tomasi Corner Detector [17]. However, the previous two methods have the limitations that the corners are only rotational invariant, when the image is scaled, they cannot be sufficiently detected. In 2004, Dr.David G. Lowe introduced the Scale-Invariant Feature Transform(SIFT) algorithm which computes the scale-invariant corner features, it computes the local maxima based on Difference of Gaussian(DoG) for difference scales and space to return a set of potential keypoints with varies scales, which achieve invariance to image scaling. Also, its computation and assignment of orientations for each keypoint achieve the invariance to image rotation [11]. Two years later, three people, Bay, HTuytelaars, T. and Van Gool, L introduced the SURF algorithm by adding lots of features upon SIFT, such as filter box for LoG approximation to improve efficiency [2]. In the same year, Roster, Edward, and Tom Drummond introduced the FAST algorithm to detect features using a 16 pixels circle, and use machine learning to train the corner detector, which makes it much faster than SURF [15]. Later, the BRIEF algorithm is introduced to reduce memory and enhance efficiency by using binary strings as an efficient feature point descriptor and comparing the Hamming distance between strings [3]. More recently, the introduction of ORB by Rublee, Ethan,et al, which builds on FAST keypoint detector and BRIEF descriptor, further improved the performance by adding orientation on FAST and the usage of rBRIEF [16]. Besides these well-known methods, there are also recent advances in feature detection to improve accuracy and efficiency. Manjunath, B. S., Chandra Shekhar, and Rama Chellappa used a scale interaction model to detect short lines, line endings, corners and any other sharp changes [12]. Van de Weijer, Joost, et al incorporated color distinctiveness into salient point detection, aiming for higher saliency with both traditional gradient strength and color saliency boosting [18]. Moreover, Robbins, Ben, and Robyn Owens described the Local energy model for 2D feature detection, features will be the local maxima in a 2D local energy map [14]. In 2006, A.E. Abdel-Hakim and A.A. Farag proposed an improved method called C-SIFT which builds the SIFT descriptiors in a color invariant space rather than gray space to represent the input image [1]. In 2009, Jean-Michel Morel and Guoshen Yu proposed an improved method called A-SIFT which simulates all image views obtainable by varying the two camera axis orientation parameters. in A-SIFT, we can reliably identify features that have undergone transition tilts of large magnitude, which is better than SIFT algorithm [13]. In 2015, Jingming Dong and Stefano Soatto create a new method called DSP-SIFT which better than SIFT in wide-baseline matching benchmarks [6].

3.2 Algorithms based on Deep Learning

Convolutional neural network with strong ability to extract details of complex features in the images, improves the traditional methods of feature detection and matching in different aspects. In 2018, D.DeTone,T.Malisiewicz and A.Rabinovich introduced a Multi-transform technique — Homographic Adaption [5] which works on computing SIFT like interest point with self-supervised training of fully-convolutional neural network [10]. The resulting detector, SuperPoint which computes both interest point and descriptors in two decoders has much better performance than LIFT and ORB. Instead of training different decoder branches independently, D2-Net method [7] introductions propose a single branch describe and detect method to extract features,which helps detect keypoints belonging to higher-level structures and unique descriptors.This helps finding image correspondences even under different image degradations.Besides SuperPoint Architecture,Z.Zhang and W.Lee propose that through utilizing graphic neural network [19] to transform feature points' coordinates into local features to avoid NP-hard assignment problems, which simplifies the further feature matching.

3.3 Feature Matching Algorithm

Recently, SIFT, SURF, ORB are successfully applied in features registration, recognition, and keypoints matching between two images. However, there are still limitations on it. When pixel intensities for two images are significantly different, there will be lots of incorrect matches of keypoints. To deal with this, Li, Qiaoliang, et al describe scale-orientation joint restriction criteria and feature descriptor refinement to exclude a huge number of incorrect matches [9]. Another feature matching method, which is introduced by Chen Yixin and James Ze Wang, represent the image by a set of segmented regions each of which is characterized by a fuzzy feature, and the resemblance of two images is defined as the overall similarity between two families of fuzzy features [4].

4 What You Plan To Do?

- Reviewing existing algorithms for feature detecting and matching, comparing their efficiency and suitability for this project
- Choosing the algorithm that is most suitable for us. Currently, the priority choice is Features From Accelerated Segment Test (FAST)
- Using the selected algorithm to detect and match the ultrasonic images and record the matching accuracy and time efficiency under the current algorithm.
- Designing some improvement based on the current algorithm, such as the scoring function for each pixel, the threshold to distinguish potential keypoints, the scanning area and the number of neighbours points for consideration.
- Comparing the accuracy and efficiency between the original FAST algorithm and the improved FAST algorithm to see the difference. Writing a report to summarize what we have achieved and what else we can work on.

5 How You Plan to Implement Your Ideas

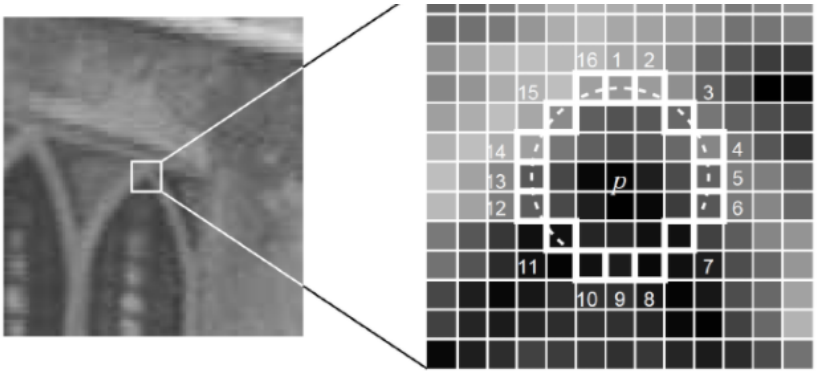
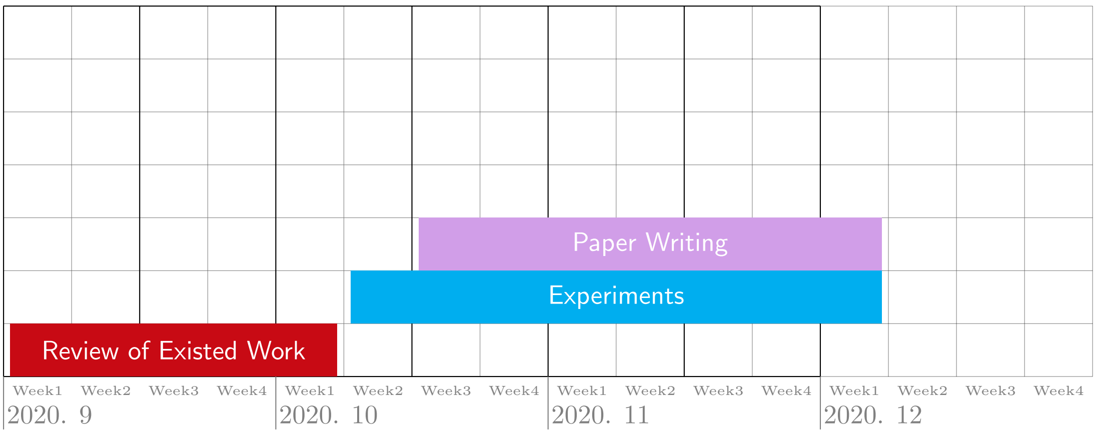


Figure 3: left:target picture,right:FAST feature detection pixel number and detection range

For this project, since the algorithms for feature matching is quite saturated, we will only focus on the improvements for the first part which is feature detection. Currently, our priority algorithm for feature detection is FAST. The potential technique for improving the FAST algorithm is shown as 3. From the image, we see that the standard to determine a corner is based on the brightness values of 16 contiguous pixels circling the potential corner. If 12 pixels of the 16 are brighter than the center of the corner by more than a threshold, then the area within the circle is a potential corner. We plan to improve the 16 points standard, doing some parameter tuning such as whether using 16 points for consideration is the best, what threshold for brightness difference is the most accurate, and number of points needed to be brighter than the center, etc. We would also like to adjust the brightness function by considering more parameters, such as difference of change in brightness for the nearby area of the experimental points, etc.

Timeline for the whole project



References

[1] Alaa E Abdel-Hakim and Aly A Farag. Csift: A sift descriptor with color invariant characteristics. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 2, pages 1978–1983. Ieee, 2006.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.

[3] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

[4] Yixin Chen and James Ze Wang. A region-based fuzzy feature matching approach to content-based image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 24(9):1252–1267, 2002.

[5] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018.

[6] Jingming Dong and Stefano Soatto. Domain-size pooling in local descriptors: Dsp-sift. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5097–5106, 2015.

[7] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8092–8101, 2019.

[8] Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988.

[9] Qiaoliang Li, Guoyou Wang, Jianguo Liu, and Shaobo Chen. Robust scale-invariant feature matching for remote sensing image registration. *IEEE Geoscience and Remote Sensing Letters*, 6(2):287–291, 2009.

[10] Yinyang Liu, Xiaobin Xu, and Feixiang Li. Image feature matching based on deep learning. In *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pages 1752–1756. IEEE, 2018.

[11] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[12] BS Manjunath, Chandra Shekhar, and Rama Chellappa. A new approach to image feature detection with applications. *Pattern Recognition*, 29(4):627–640, 1996.

[13] Jean-Michel Morel and Guoshen Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM journal on imaging sciences*, 2(2):438–469, 2009.

[14] Ben Robbins and Robyn Owens. 2d feature detection via local energy. *Image and Vision Computing*, 15(5):353–368, 1997.

[15] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006.

[16] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

[17] Jianbo Shi and Carlo Tomasi. Good features to track in: Conference on computer vision and pattern recognition. CVPR, 1994.

[18] Joost Van de Weijer, Theo Gevers, and Andrew D Bagdanov. Boosting color saliency in image feature detection. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):150–156, 2005.

[19] Zhen Zhang and Wee Sun Lee. Deep graphical feature learning for the feature matching problem. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5087–5096, 2019.