영상에서 객체 특성을 이용한 거리 측정 알고리즘

왕동악*, 황보택근**
*가천대학교 전자계산학과
e-mail: wangdongyue89@126.com

Algorithm for depth calculation using object characteristics in an image

Dong-Yue Wang*, Taek-Guaen Whang-Bo**
*Dept of Computer Science, Gachon University

Abstract

Mobile phone GPS positioning system is a function we often use, which can make user locate his own position all time. But due to lots of inconvenience brought to us by some error existing in mobile phone positioning system, how to accurately position becomes a hot research topic at home and abroad. This paper puts forward a way of precise calculating position. Firstly, take object's image with mobile phone camera; then, process the image to extract and identify object inside; finally, calculate the distance between the object and user with the object's information stored in the database.

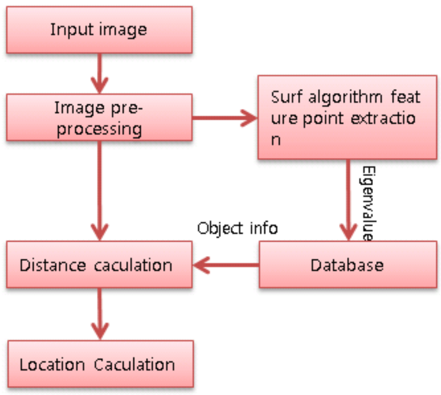
1. Introduction

Almost every mobile phone is equipped with GPS positioning system, which can make user locate his own position all time. but positioning system is just a small part of mobile phone function, and positioning has a certain error (usually from 30 to 50 meters according to measurement), which brings a lot of inconvenience to user, so accurate positioning has become a hot research topic at home and abroad, and a large number of auxiliary positioning systems have also emerged, such as AGPS, etc. This paper puts forward an auxiliary positioning method based on object's additional information. Firstly, obtain the real-time image taken by mobile phone camera; then process and identify the image obtained; finally calculate the distance between the object and user with the object's information stored in database, and calculate user's own position. The experimental result shows that error of this method is about 2m, compared with actual measurement result. In the future, we will decrease the error to less than 1m.

This research is supported by Ministry of Culture, Sports and Tourism(MCST) and Korea Creative Content Agency(KOCCA) in the Culture Technology(CT) Research & Development Program [R2012030006].

2. System Composition

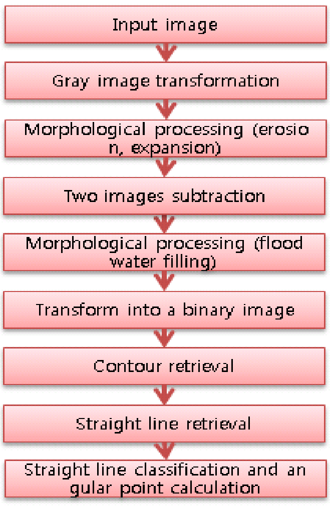
This system operates on mobile devices with Android System. Firstly, obtain the real-time image with mobile phone camera, and frame and identify the image on mobile phone; then, extract the image identification information from database (actual height of the image, GPS information, etc.); next, calculate actual distance between user and the object based on internal parameters in mobile phone camera; finally, calculate user's position based on the actual distance. The system composition diagram as shown in the following diagram (Fig. 1).



(Fig. 1) System Composition Diagram

2.1 Image Pre-processing

Image pre-processing is a very important process. For the convenience of measuring object's additional information, we use rectangle object as basic reference. That means that 4 vertices of the object needs to be found when determining the object's size in the image. The method used in this paper is shown in the following diagram (Fig. 2).

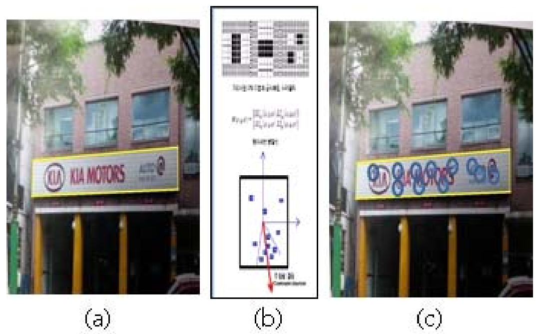


(Fig. 2) Flow Diagram for Finding Image Angular Points

Firstly, transform colorful image into gray one, and then conduct morphological processing (erosion and expansion) to it to eliminate noises and some fine lines inside. Secondly, substract the image obtained after morphological processing by the original gray image input. Because the image by morphological processing has been erosed, the object's image inside is smaller than that in the original one, and the object's outline will appear after subtraction. Thirdly, the image after subtraction will be processed by the way of flood water filling. This system takes image's central point as reference point, which will be processed by the way of flood water filling. And the image central point will be at any object which will be intended for flood water filling processing. Fourthly, separate the object after flood water filling processing from its image by processing it with binarization. Fifthly, find the outline of the object by the way of outline detection before conducting straight line detection to it. Affected by noise, the outline is not some straight lines, so many straight lines will be detected, which requires a classification to these lines. Also, because the object is rectangle, the lines already detected will be classified into 4 categories according to 4 sides of a rectangle, among which a straight line will be selected for calculating their intersection point, respectively. The calculated 4 intersection points are the object's 4 angular points.

2.2 Object Identification

Because distance calculation will be done after reading the object's information from database, so we need to identify the object firstly. The common algorithms in extracting and matching the feature points of the object are SIFT (Scale-invariantfeaturetr ansform) and SURF(Speed-Up Robust Features). SIFT algorithm can accurately detect the feature points, but it is slower in operation due to its bigger calculating, while SURF algorithm overcomes the shortcoming of SIFT's slow calculating speed. This system adopts SURF algorithm, with its retrieval way shown in the following diagram (Fig. 3).



(Fig. 3) SURF Algorithm Feature Points Retrieval

(a) is the object's area obtained in pre-processing after image input. After that, SURF algorithm is used to retrieve its feature points in this area.

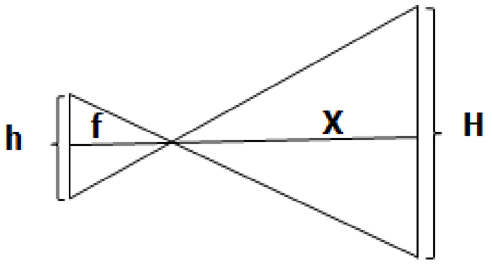
(b) is the operational schematic diagram of SURF algorithm, with Hessian matrix as its core. Its algorithm as shown in formula (1).

$$H(x,y,\sigma) = \begin{bmatrix} LI_{xx}(x,y,\sigma) & LI_{xy}(x,y,\sigma) \\ LI_{xy}(x,y,\sigma) & LI_{yy}(x,y,\sigma) \end{bmatrix} \qquad (1)$$

(c) is the feature points detected with SURF algorithm. Match these feature points with those stored in database, then we can obtain the object's additional information from the database based on the identified object (additional information: actual height and distance between both ends of the object).

2.3 Distance Measurement

Next step after obtaining the object's additional information is to calculate distance according to its information. In this paper, we caculate the distance according to the object's actual size and its size scale in the image, with its measurement principle shown in the following diagram (Fig. 4)



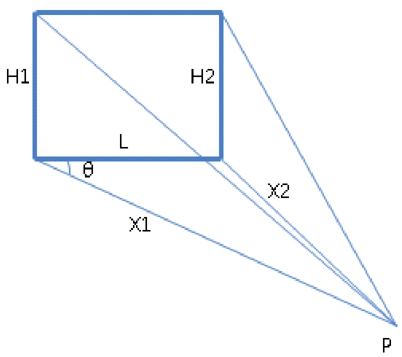
(Fig. 4) Distance Calculation Principle Diagram

"f" is camera's internal parameters obtained by camera calibration; "h" is the object's height in its image, which has been calculated at the stage of image pre-processing; "H" is the object's actual height, which is obtained from database at the stage of object identification; "X" is the distance between the object and user, which can be calculated with triangle similarity principle (See formula 2).

$$\frac{h}{f}=\frac{H}{X}\Rightarrow X=\frac{H*f}{h} \qquad (2)$$

2.4 Position Calculation

After distance calculation is completed, next step is to calculate user's position based on the object's position. we take both ends of the object as reference point to calculate user's position, with its calculation principle shown in the following diagram (Fig. 5).



(Fig. 5) Position Calculation Principle Diagram

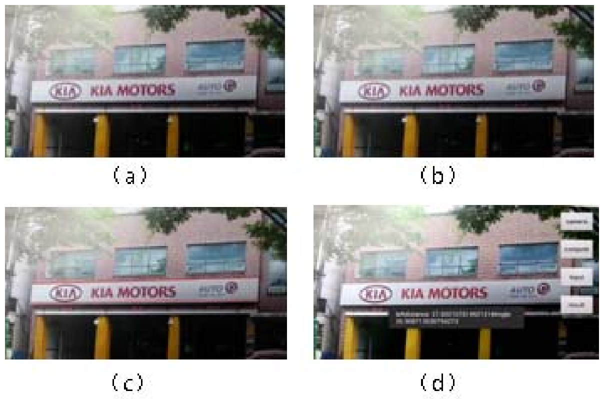
"H1" is the object's actual height on the left side; "X1" is the calculated distance between user (P) and the object's left side. In the same way, "X2", the distance between user (P) and the object's right side, can be calculated according to the object's height on the right side; "L" is obtained from database. And angle "Θ" can be caculated with triangle cosine theorem formula (See formulat 3).

$$\cos\theta=\frac{X1^2+L^2-X2^2}{2\times X1\times L} \qquad (3)$$

Finally, we use the distance between user and the object and angle "Θ" to output user's position.

3. Test Result and Future Research Plan

One application program is completed on Samsung GALAXY S3 for testing results of the system which this paper puts forward. The test pictures as shown in the following (Fig. 6).



(Fig. 6) Operational Pictures

"a" is the mobile phone's preview picture; "b" is the picture taken by the mobile phone. Meantime, user stores the picture in the mobile phone, and locks and processes its preview picture processed; "c" is the picture of pre-processing. User firstly reads the taken picture from mobile phone, and then find 4 angular points of the object through pre-processing the picture; "d" is the picture showing user's position. User can seperate 4 angular points of the object after pre-processing the picture, and then extract its feature points, before matching them with those in database to identify the object. The object's additional information (left and right end height of the object, and the distance between both ends) will then be extracted after its identification After that, the distance will be measured and the position calculated. Upon outputing user's position data, we output the distance and angle, with the object's left end as benchmark. The test results as shown in the table below.

<table 1>Test Result

Test	Left Height	Right Height	Distance from left side(Measured)	Distance from right side(Measured)	Distance from left side(Caculated)	Distance from right side(Caculated)
Test1	5.25m	5.25m	10m	10m	10.6m	10.8m
Test2	5.25m	5.25m	15m	18m	16.5m	18.2m
Test3	6.8m	6.8m	25m	20m	24.2m	25.7m
Test4	10m	8m	20m	25m	20.3m	25.6m
Test5	10m	8m	30m	35m	31.2m	37m
Test6	8m	8m	20m	15m	20.2m	15.8m
Test7	5m	5m	10m	12m	10.1m	11.9m
Test8	3m	4m	15m	18m	14.5m	18.2m
Test9	3m	4m	20m	15m	18.3m	14m
Test10	5m	5.5m	25m	27m	26.2m	27.8m

Some error exists between actual measurement data and those measured by the system. The reasons for that as follows: Not accurately finding 4 angular points of the object due to illumination of light, as well as tilt- ing mobile phone when taking pictures, for which we will research how to eliminate the error and make a more accurate positioning in the future.

References

[1] 한종원, 조진수, 이일병, “단안단서를 이용한 자연영상의 상대적 깊이지도 생성”, 한국컴퓨터종합학술대회 논문집 Vol.33, No.1(B), pp.368-369, 2006.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars and Luc BanGool, "Speeded-Up Robust Features(SURF)", Computer Vision and Image Understanding, Vol.110, pp.346-359, 2008.

[3] David G.Lowe,"Object Recognition from Local Scale-Invariant Features", Computer Vision 1999, The Proceedings of the Seventh IEEE International Conference on,Vol.2, pp.1150-1157, 1999.

[4] DAVID G.LOWE, "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision 60(2), pp.91-110, 2004.

[5] Yan Ke, Rahul Sukthankar, "Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on", Vol.2, pp.Ⅱ-506-Ⅱ-513, 2004.

[6] 최동석, “CML을 이용한 물체의 거리측정” 성균관대학교 석사졸업논문, 2009