Jan KLIMASZEWSKI, Barbara PUTZ
Warsaw University of Technology, Warsaw

RECURSIVE TECHNIQUES FOR FAST STEREOVISION WITH LOW QUALITY IMAGES IN MOBILE ROBOTICS

Key words

Stereo matching, area based, recursion, box filtering, object segmentation.

Summary

One of the main problems in mobile robotics is obtaining knowledge about the surroundings from sensor data. This article describes attempts of fast 3D observed scene feature extraction based on information from a two-camera stereovision system. The additional assumption is that the robot vision system, dedicated to the navigation purpose, should be able to work with low quality images. The power of recursive techniques in the implementation of real-time working algorithms is presented in regard to standard area-based stereo matching but mainly focused on the new recursive algorithm for characteristic object segmentation in low quality images.

Introduction

The idea of stereovision algorithms is inspired by the functionally excellent visual system used by a huge amount of the animal kingdom. Most of the highly developed animals (among which humans) use stereovision to recognise their surroundings and are able to perfectly solve SLAM (Simultaneous Localisation and Mapping) problems.

We made two basic assumptions regarding to design and functionality purpose of a stereovision system:

– The vision system should manage to provide 3D information about a scene based on low quality images. Calibration of such a system should not be the main problem.
– The functionality of the vision system is dedicated to the navigation purpose, i.e., the emphasis should be put on detecting the localisation of objects in the mobile robot's surroundings.

Low quality images will lead to sparse 3D information; however, it could be enough for the purpose of navigation.

1. Methods to increase the computation efficiency

Stereo matching algorithm produces a dense disparity map for matched regions in left and right images. The disparity map may then be used to compute dense depth maps of the scene points [1, 2]. A fast stereovision algorithm designed by Changming Sun [3, 4] and many others [1, 5–8] contain a few types of methods for computations speed-up.

The classical methods of stereo matching are the intensity based area-correlation techniques. Direct calculation of the cross-correlation coefficient for the correlation window of size W = (2K+1)(2L+1) has nearly (2K+1)(2L+1) redundancies. To avoid it, a recursive box-filtering technique has been proposed [1, 3, 4]. We present the idea of this technique for calculating the mean value of pixels intensities within the window (Fig. 1).

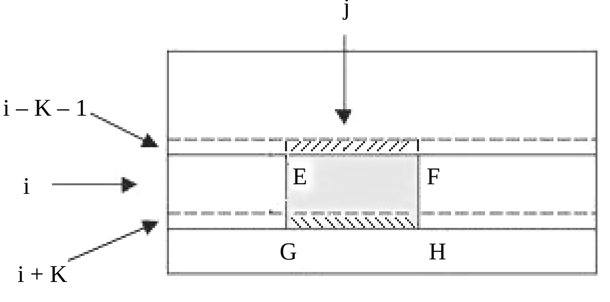


Fig. 1. The box filter: a window EFGH with $(2K+1)$ rows and $(2L+1)$ columns, centred at (i, j), for which the mean value of pixels intensities is recursively calculated

The mean value $F_{ij}=1/W\sum_{m=i-K}^{i+K}\sum_{n=j-L}^{j+L} f_{ij}$ of pixels intensities f_{ij} within the window centred at pixel (i,j) can be calculated as $F_{ij}= S_{ij}/W$, where S_{ij} is obtained using recursive relation (Fig. 1):

$$S_{ij} = S_{i-1,j} + B_j(i+K) - B_j(i-K-1) \tag{1}$$

where the buffer $B_j(i) = \sum_{m=j-L}^{j+L} f_{im}$. The explicit initialisation of $S_{iL.}$ is necessary.

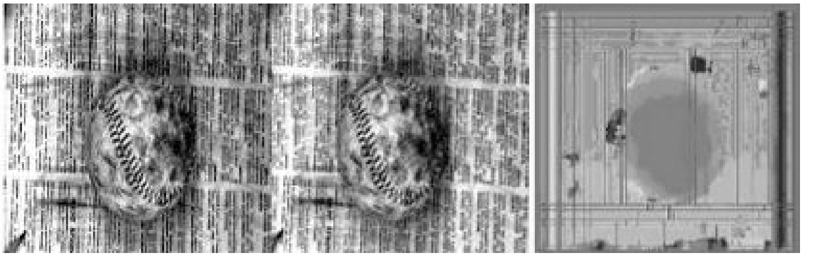
To calculate the values of the buffer $B_j(i)$, similar recursion idea may be used:

$$B_j(i) = B_j(i\text{-}1) + f_{i,j\text{-}L} - f_{i,j\text{-}L\text{-}1} \tag{2}$$

This recursive method of F_{ij} calculation is very fast (a few operations for each output pixel) and independent of box size. In the same recursive way, the pixel variance within the box and finally the normalised cross correlation (NCC) of two local windows of both images can be quickly calculated. For computational efficiency, the recursive calculation of sum of absolute differences (SAD) instead of NCC can also be used [3, 4]. A method of paralleling of the SAD recursive procedure and its VLSI implementation can be found in [6], as well as an interesting processing scheme for the dual-core architecture of the embedded processor, with correlation-based matching using three levels of recursion [7].

Other techniques for fast calculations, proposed in [3, 4], have also been implemented: coarse-to-fine (CTF) scheme with multi-resolution pyramid data structure and rectangular subregioning of images at different levels of the pyramid. The CTF scheme used by many other authors [5, 8], enables the exclusion of the useless stereo match of images just at the start of the analysis process.

a)



b)

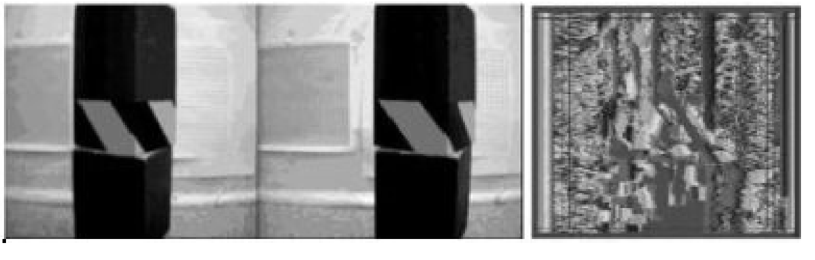


Fig. 2. Results of the implementation of the Changming Sun's algorithm: a) left – two good-quality stereo-pair images [9]; right – the computed disparity map (rectangular subregions are marked by lines); b) the same for medium quality images

Sample results of implementation of the above recursive and other fast techniques are presented in Fig. 2. As we can see, the algorithm is good for good quality images but does not provide clear information for images derived from medium or low quality hardware.

2. Methods used for low quality images analysis

Experiments described above with stereovision algorithms have led to an observation, that pixel-by-pixel stereovision algorithms are not effective for low quality images analysis. Another observation can be made that, while it is relatively easy to point out corresponding objects on the left and right stereo-pair images, pointing out corresponding pixels is a much harder task. The next part of studies shows usage of two basic image analysis methods used for characteristic objects contained by the observed scene.

The first step of the analysis is the edge detection. This is intended to cut out the computationally intensive feature analysis and allow dealing with low quality images (high level of noise). It has been solved by filtering with a LoG (Laplacian of Gaussian) filter. The Laplacian filter is commonly used for edge detection in image analysis, but it has a side effect of increasing the noise level. This side effect is removed with Gaussian filtering. A combination of these two techniques results in an efficient edge detection algorithm. The resulting binary image is used for further analysis. The next step is to segment the image regarding to the objects placed in the observed scene. For this purpose, recursive image segmentation was designed. Algorithms developed for this purpose are presented below.

The next step of the analysis is the image object segmentation (information about segmentation methods can be found in [10]). Methods of amplitude segmentation like bilevel/multilevel luminance thresholding and a bilevel/multilevel colour component thresholding were considered not effective because of the complexity of analysed images and because of a possibility of different objects having the same colour or luminance. Clustering segmentation methods (including texture segmentation) were considered too computationally intensive for real-time application; furthermore, the segment distinguishing features are not known.

Based on the above, only region growing segmentation or boundary segmentation could be used to segment the data properly. To solve the presented problem, the region growing segmentation was chosen because of its computational simplicity. Furthermore, the region growing with pixel amount threshold is able to deal with noisy data.

In the beginning, the region growing segmentation algorithm is used to segment binary image received in LoG filtration. The segmentation algorithm design should result in an algorithm able to segment transformed binary images for detecting objects represented by clusters of pixels larger than a given threshold (amount of neighbouring pixels, in algorithms description represented as *prog* constant), as well as groups of smaller objects located near each other. In other words, the algorithm should be able to detect patterns built from smaller objects not connected with each other. These patterns are easily detected by human vision, however the standard region-growing algorithm cuts them out because of an insufficient amount of pixels corresponding to each small object.

3. Algorithms presentation

Main algorithm

This is the entry algorithm for the whole recursive segmentation. In this algorithm, a decision is made about the kind of segmentation to use and whether the segmentation should be executed for small (below threshold pixel amount value) or large objects.

1) Set flag binary image *F* to 0.
2) For each pixel (*x*,*y*) in analysed image *O* do the following:
   - If (*F*(*x*, *y*) = = 0 and *O*(*x*, *y*) = = 1) then
     – Call *Add1*(*x*, *y*).
     – If (amount of pixels contained by *list1* < *prog*) then
   - Call *Add2*(*xsr*, *ysr*).
     – Create an object from pixels contained by *list1* and reset *list1*

where:

x, *y* – pixel coordinates,
O(x,y) – analysed image – binary image with detected edges after LoG filtering,
F(x,y) – flag image of the same size as analysed image, each pixel contains value 0 or 1,
list1, *list2* – list structures for containing pixel coordinates
xsr, *ysr* – centre of pixel group contained by *list1*.

***Add1*(*x*, *y*) function definition**

1) Set flag image pixel *F(x, y)* to 1.
2) Add (*x*, *y*) pixel to *list1*.
3) For each neighbouring pixel do the following:
   - If (*F* = = 0 and *O* = = 1) then
     – Call *Add1* for these coordinates.

This is a recursive function similar to the region-growing algorithm. Neighbouring pixels are being added to *list1*. When the recursive calling is finished, *list1* contains pixels pertained to one group not connected with any other pixel.

***Add2*(*xsr*, *ysr*) function definition**

1) For each pixel in the radius *r* surroundings of (*xsr*, *yxr*) do the following:

- If (F = = 0 and O = = 1) then
  – Call *Add3* for these coordinates.
  – If (number of pixels on the *list1* < *prog*) then
    - Add pixels from *list2* to *list1* and reset *list2,*
    - Call *Add2* for the centre of pixels contained by *list2* coordinates.

  – Else:
    - Create an object from pixels contained by *list2* and reset *list2*.

This is not a recursive function either. Its main functionality is finding small objects localised near each other that could not compose one object when isolated. They are detected as one "spread" object. When the recursive calling of *Add2* is finished, one single object is being created from pixels contained on *list1* (last step in the main algorithm). When a larger object is encountered while grouping smaller objects, pixels that create this larger object are grouped on *list2*. The aim of this is the separating pixels from small and large objects.

***Add3*(*x,y*)** function is analogical to *Add1*. The main difference is that the found pixels are added to *list2* instead of *list1*.

Algorithms presented above give good results for non-connected objects. Results are shown in Fig. 3.

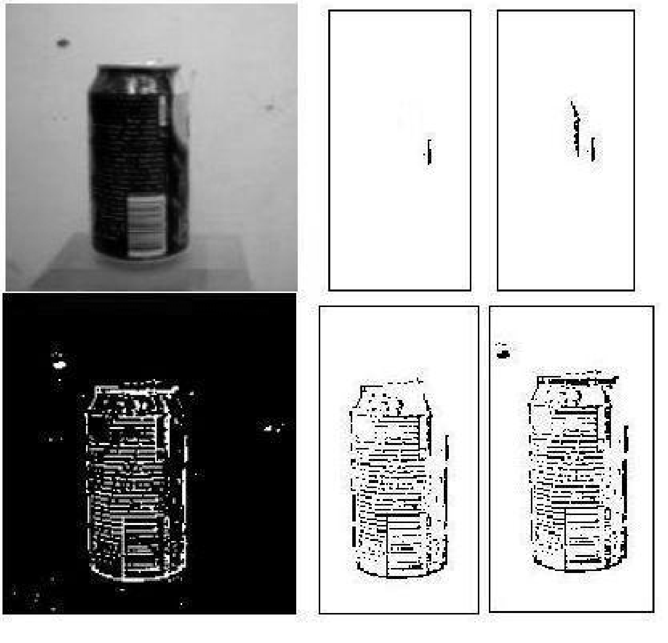


Fig. 3. Results of LoG filtration and recursive segmentation. Left: input image (above) and image after LoG (below). Right: exemplary figures illustrating the work of segmentation; the result of adding successive objects: the first two (above) and the last two (below)

It should be noted that a similar approach and usage of recursive techniques as presented here can be found in [11]. While preparing the image for further segmentation matching, it is important to preserve all characteristic image features, i.e. some objects that could be matched in two stereo-pair images as the same. The method presented in [11] is used to remove so called "unwanted data"; it preserves a smaller amount of image details, and it seems insufficient for stereo image object segmentation to be solved with assumptions presented in the introduction.

Conclusions

The usage of stereovision system in regard to the localisation of objects in a robot's surroundings brings advantages, such as relatively low financial cost (e.g. compared to a laser scanner).

The literature analysis and presented results of experiments of a few algorithms show that the main problem of creating a stable and robust stereovision system is not the time of working with stereovision algorithms, which can be reduced even using software methods. Also, hardware platforms are being more and more effective and the financial cost is not so high.

The proposed new method is based on the region growing method. This method maintains the region growing ability to simultaneously segment and to deal with noisy data. Furthermore, it extends the region growing capabilities by preserving the poorly visible image texture details (instead of classifying them as a noise). The method does not need any additional noise removal algorithms contrary to some other segmentation methods like [12]. Because of the above, the presented method is ideal to deal with noisy data sets.

The algorithms need further work and have to be completed to increase segmentation quality. Two different objects sharing a common edge on the image need to be separated, and this functionality is necessary to add.

Another problem that has to be solved is stereovision system camera calibration. The important problem while working with stereovision system is the changing of hardware parameters (and cameras de-calibration) during movement of the mobile platform. Stereovision system based on object detection on low quality images should be working better after addressing the above problem.

Scientific work financed partly by Rector of Warsaw University of Technology within the WUT Programme "Heterogenic net of collaborating mobile robots", Warsaw 2007.

References

1. Brown M.Z., Burschka D., Hager G.D.: Advances in computational stereo. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003, 25, 8, 993–1008.
2. Hartley R., Zisserman A.: Multiple view geometry in computer vision. Cambridge University Press, 2003.
3. Sun Ch: Multi-resolution Rrectangular subregioning stereo matching using fast correlation and dynamic programming techniques. CMIS Report No. 98/246, 1998, 24.
4. Sun Ch: Fast stereo matching using rectangular subregioning and 3-D maximum-surface techniques. International Journal of Computer Vision, 2002, 47, 1/2/3, 99–117.
5. Franke U., Joos A.: Real-time stereo vision for urban traffic scene understanding. Proc. of the IEEE Intelligent Vehicles Symposium 2000, Dearborn (MI), USA , 2000.
6. Miura K., Hariyama M., Kameyama M.: Recursive computation-based stereo matching and its implementation in VLSI. Electronics and Communications in Japan, 2004, 87, 12, 19–27.
7. Khalegi B., Ahuja S., Wu Q.M.J.: A new miniaturized embedded stereo-vision system. Canadian Conf. on Computer and Robot Vision, 2008, 26–33.
8. Sizintsev M., Wildes R.P.: Efficient stereo with accurate 3-D boundaries. Proc. of the British Machine Vision Conf., 2006, 237–246.
9. http://extra.cmis.csiro.au/IA/changs/stereo – Fast Stereo Matching Demo, by Changmin Sun.
10. Pratt W. K.: Digital image processing.. Wiley-Interscience, 2007.
11. Sun W., Spackman S. P.: Multi-object segmentation by stereo mismatch. Machine Vision and Applications, DOI 10.1007/s00138-008-0127-1, 2008.
12. Gao L., Jiang J., Yang S.Y.: Constrained region-growing and edge enhancement towards automated semantic video object segmentation. J. Blanc-Talon et al. (eds.): Springer-Verlag, Berlin Heidelberg, 2006, 323–331.

Reviewer:
Andrzej DZIELIŃSKI

Rekurencyjne techniki szybkiej stereowizji dla obrazów niskiej jakości w robotyce mobilnej

Słowa kluczowe

Stereoskopowe dopasowywanie obszarami, rekurencja, segmentacja obiektów.

Streszczenie

Artykuł opisuje próby ekstrakcji cech trójwymiarowych obserwowanej sceny na podstawie informacji, pochodzącej z dwukamerowego systemu stereowizyjnego robota mobilnego. Jako dodatkowe założenie przyjęto, że system wizyjny robota, przeznaczony do celów nawigacji, powinien działać bazując nawet na obrazach niskiej jakości, która często jest efektem braku sztywności układu kamer podczas ruchu robota. Tematem wiodącym jest ukazanie siły technik rekurencyjnych, pozwalających na implementację algorytmów pracujących w czasie rzeczywistym.

Zaprezentowano zasadę stosowania rekurencji do eliminacji obliczeń Redundantnych w algorytmach stereoskopowego dopasowywania obszarami oraz nowy algorytm rekurencyjny opracowany w celu wychwytywania obiektów charakterrystycznych w obrazach niskiej jakości. Algorytm, po wykonanej filtracji LoG, pozwala na wykrywanie obiektów dużych oraz grup małych obiektów, zależnie od ustawionego progu. Ponadto, zastosowanie algorytmu nie wymaga użycia dodatkowych metod usuwania szumów z obrazu. Opracowana metoda, przeznaczona do przygotowania obrazów stereowizyjnych do dalszej analizy, pozwala zachować w postaci obiektów większą ilość charakterystycznych cech widocznych na obrazie. Wyodrębnione obiekty mają umożliwić dalszą rekonstrukcję sceny 3D.