Transactions of the VŠB – Technical University of Ostrava, Mechanical Series
No. 1, 2010, vol. LVI
article No. 1766

Petr NOVÁK, Petr ŠPAČEK, Vladimír MOSTÝN*

METHODS FOR FINDING EQUAL POINTS IN THE IMAGES FOR STEREOVISION

METODY NALEZENÍ STEJNÝCH BODŮ V OBRÁZCÍCH PRO STEREOVIZI

Abstrakt

Příspěvek popisuje tři způsoby hledání identických bodů ve dvou obrazech – levého a pravého – pro potřeby stereovize. Jsou použity metody: vzájemná korelace, "graph-cut" a SAD (Sum of Absolute Differences). Operátor má nasazeny 3D brýle a scéna je snímaná dvěma rovnoběžným kamerami. Za použití vyvinutého software, je provedena transformace obrazu tak aby každému oku příslušel vhodně posunutý obraz.

Abstract

The article describes three methods finding the identical points in images pair – left and right – for sterevision. Thera are used methods: cross-correlation, graph-cuts and last one SAD - Sum of Absolute Differences. The operator has 3D helmet putted on and the scenery is taken by pair of parallel cameras. Using the developed software, the image transformation is done. Every eye has suitable shifted image.

1 INTRODUCTION

This paper describes three possible methods for finding matching points in the image. Knowlwdge of the position of these points is important for stereovision systems – 3D vision. Depending on it, the both - left and right image will be reciprocally shifted. The first procedure is based on cross-correlation, second one on graph-cut (based on energy minimization) and last one on SAD - Sum of Absolute Differences. All methods, described here, use preprocessed images (from left and right cameras respectively) - after cameras calibration and their rectification.

2 CROSS-CORRELATION

The task is to find place of mask (obtained for examle as subimage *w(x,y)* of right image) in the second image (in our case it will be the left image). Be using of the normalised cross-correlation between the mask and left image, this place – if exists – will be characterized by maximal value of the normalised cross-correlation coeficient. In our case it will be to tend 1 [1][11].

The expression for compute of the normalised cross-correlation is:

$$\gamma(x,y) = F(f,w) = \frac{\sum_{s,t}[w(s,t)-\bar{w}]\left[f(x+s,y+t)-\overline{f_{xy}}\right]}{\sqrt{\sum_{s,t}[w(s,t)-\bar{w}]^2 \sum_{s,t}\left[f(x+s,y+t)-\overline{f_{xy}}\right]^2}} \quad (1)$$

where: w is mask, $\bar{w}$ is mean value of the elements of the mask, f is image and $\overline{f_{xy}}$ is the its mean of in the region under the mask. The summation is taken over the values of s and t such the region under the mask. The values of $\gamma(x,y)$ are evaluated in <-1, 1> interval. A high value of normalised cross-correlation (*normalised cross-correlation in absolute expression in <0;1> interval*) means a good correspondence between the mask and the founded place in the image.

* prof. Dr. Ing. Petr Novák, Ing. Petr Špaček, prof. Dr. Ing. Vladimír Mostýn, VSB-Technical University of Ostrava, Department of Robotics, 17.listopadu 15/2172, 70833 Ostrava-Poruba, Czech Republic, +420 596 993 595, petr.novak@vsb.cz

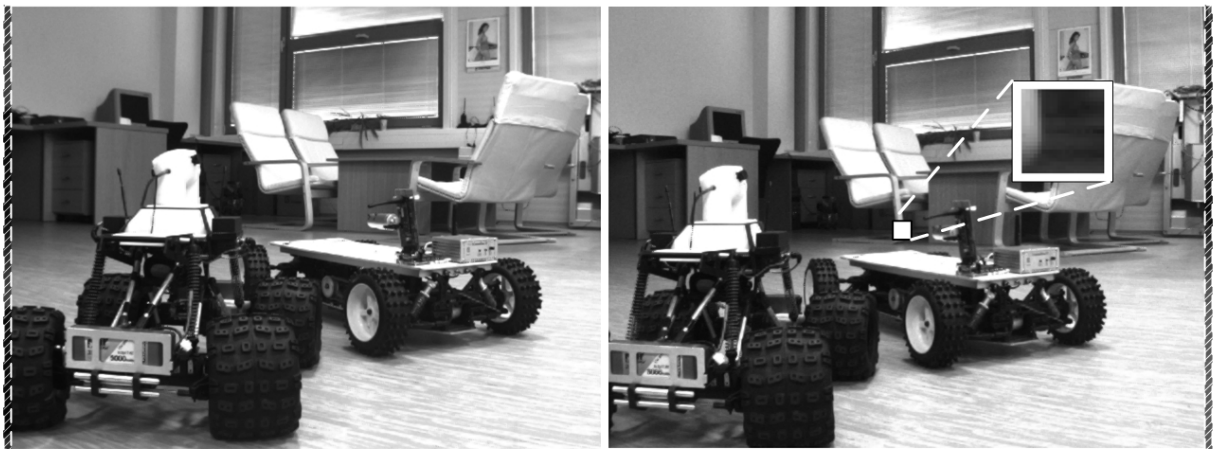

Fig. 1 Left and right image – 640 x 480 pixels, after images preprocessing: cameras calibration, rectification and converting both original RGB images to grayscale. (The meaning of the dashed parts is explained in the text.)

At first, the mask *w* for matching has to be established. It is directly cropped from the central part of right picture, in our case in size 21 x 21 pixels - see zoomed detail in the right image Fig. 1. Its x position is *xmin*. Next step should be finding position mask in the left image. Because both images are rectificed, it is sure, that finding position in the left image must lie in central zone too - if exists. Hence, the expression (1) is applied only on central rows - 640* x (5+21+5) pixels - see Fig. 2. Its number is high of mask *w* plus five rows above and five rows below (elimanation of the case of the inaccurate rectification). The found position of best correlation is *xpeak* and it is marked by quarter in the Fig. 2.

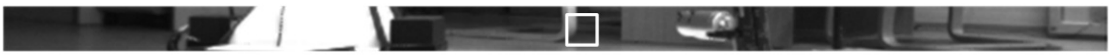

Fig. 2 Cropped image - from the central part of the left image in the Fig. 1. (Size: 640 x 31 pixels). The square labels found place with highest cross-correlation coefficient.

Fig. 3 Visualisation of the normalised cross-correlation $|\boldsymbol{\gamma}(\boldsymbol{x},\boldsymbol{y})|$ between cropped image *f* and mask *w* . (Place with "top" value - 0.982 - is marked by arrows.) Size: 660 x 51 pixels – incresed by mask *w*.

The visualisation of the normalised cross-correlation $|\boldsymbol{\gamma}(\boldsymbol{x},\boldsymbol{y})|$ (1) of the part left image and mask is illustrated in the Fig. 3. The position of the highest cross-correlation coefficient is marked by pair of arrows. In this case, its value is 0.982. The similar visualisation of the (1) - now in 3D view - is presented in the Fig. 4.

* The additional optimilization there is possible too – searched points will not be situated in the left and right borders.

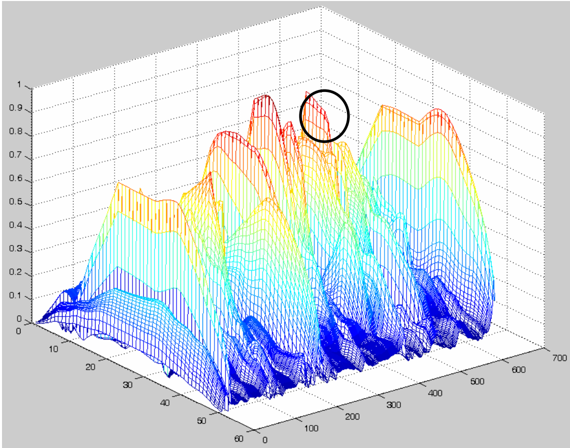

Fig. 4 3D visualisation of the $|\boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y})|$.

The final step there is shift left and right images together. It is achieved by trim of the left part of the left image and right part of the right image. The width of each trim is same and can be calculated as:

$$\frac{xpeak - xmin}{2} \tag{2}$$

where:
xmin is original position of the mask in the right image in *x* axis,
xpeak is position with the best equivalence of the mask in the *x* axis of the left image.

Output pair of images has smoler sizes in x axis – see dashed parts in left and right sides in the Fig. 1. Those images are screened on displays of 3D helmet and can be coupled with others text and/or graphical informations.

2.1 Mask choosing

Described procedure uses (for simplicity) mask cropped from fixed position in the centre of the right image. For real application it is necessary to has mask cointaining as much brightness levels as possible. For example in the case of mask cointained monochromatic subject (part of wall, cupboard, curtain, etc.) will be problem with matching corresponding points in the second image. Respectively, the cross-correlation will not contain sharp peak and determination of the best correspondence will be problematic or impossible. For this reason it is necessary in real utilisation to take care of mask automatic selection.

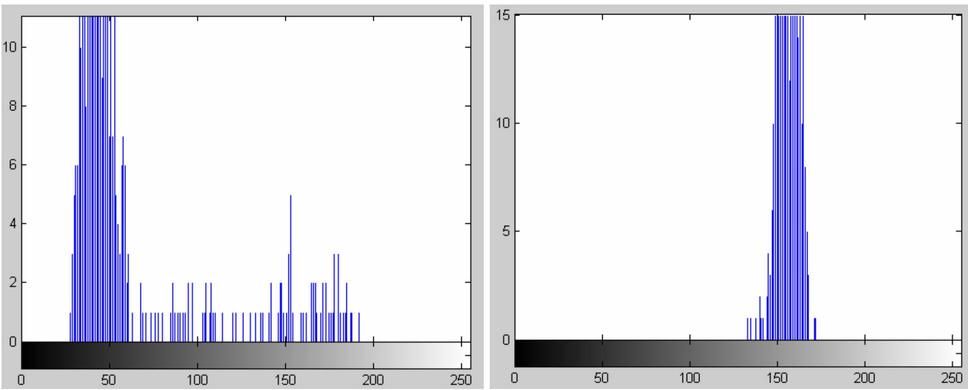

Fig. 5 The histograms of the mask acquired from the centre of right image in Fig. 1 (left) and histogram of mask cropped in backrest of armchair the same right image.

3 FINDING OF STEREO CORRESPONDENCE USING GRAPH-CUT

The next method, which can be used to find stereo correspondence, is "graph-cut". This algorithm is based on energy minimization. Ideally, a pixel in one image should correspond to at most one pixel in the other image, and pixels that correspond to no pixels are labelled as occluded. We refer this requirement as *uniqueness*. We address the correspondence problem by constructing a problem representation and an energy function, such that a solution which violates uniqueness will have infinite energy.

3.1 Energy function

Now we define energy function according to [2].

$$E(f) = E_{data}(f) + E_{occ}(f) + E_{smooth}(f) \tag{3}$$

Where

- a data term $E_{data}(f)$, is results from the differences in intensity between corresponding pixels
- an occlusion term $E_{occ}(f)$ imposes a penalty for making a pixel occluded
- a smoothness term $E_{smooth}(f)$ makes neighbouring pixels in the same image tend to have similar disparites.

The data term will be $E_{data}(f) = \sum_{a \in A(f)} D(a)$ typically for an assignment a = p,q>, $D(a) = (I(p) - I(q))^2$, where I gives the intensity of the pixel.

The occlusion term imposes a penalty C_p if the pixel p is occluded. We will write this as this equation.

$$E_{occ}(f) = \sum_{p \in P} C_p \cdot T(|N_p(f)| = 0) \tag{4}$$

And smoothness term is defined by this equation

$$E_{smooth}(f) = \sum_{\{a1,a2\} \in N} V_{a1,a2} \cdot T(f(a1) \neq f(a2)) \tag{5}$$

Detailed deduction can be found in [2].

3.2 Expansion move algorithm

This algorithm is to efficiently minimize E with the smoothness term among all unique configuration using graph-cut. The output of this method will be local minimum. In particular, consider an input configuration of pixel pairs *f* and disparity α. Another configuration f' is defined to be within a single α-expansion of *f* if some assignment in *f* become inactive and some assignment with disparity α become active.

The steps of expansion algorithm is.:

1) Start with an arbitrary unique configuration *f*
2) Set success := 0
3) For each disparity α
   a) Find f'' = argmin E(f') among unique f' within single α-expansion of *f*
   b) If E(f'') < E(f), set f := f'' and success := 1
4) If success = 1 goto 2
5) Return *f*

3.3 Graph cuts

Let $G = \langle \mathcal{V}, \varepsilon \rangle$ be weighted graph with two distinguished terminal vertices *{s,t}* called the source and sink. A *cut* $C = \mathcal{V}^s, \mathcal{V}^t$ is a partition of the vertices into two sets that $s \in \mathcal{V}^s$ and $s \in \mathcal{V}^t$. The cost of the cut, denoted *C*, equals the sum of the weights of the edges between a vertex in $\mathcal{V}^s$ and a vertex in $\mathcal{V}^t$.

The minimum cuts problem is to find the cut with the smallest cost. This problem can be solved very efficiently by computing the maximum flow between the terminals.

3.4 Results of graph cuts method

The graph cut method was tested on images, which were acquired by two parallel cameras. Images from cameras were rectified before they were used. Images from left and right camera are showed on **Fig. 1**.

From image the disparity was computed. From this disparity were made a graph that shows the results – see

Fig. 6.

From graph we can see that the disparity, which is in higher part of spectrum, is not correct. In these pixels, with incorrect disparity, algorithm doesn't find correct disparity or pixels may be occluded. For all that the algorithm is successfully in computing correspondence, but very time-consuming.

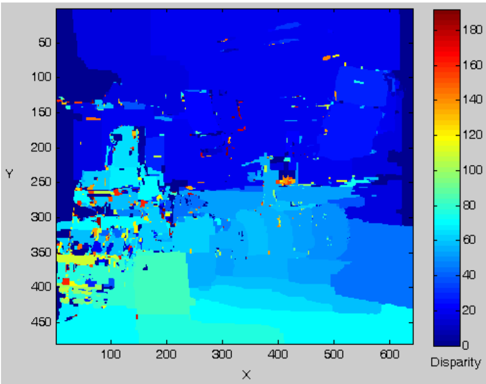

Fig. 6 Graph of computed disparity - graph cuts

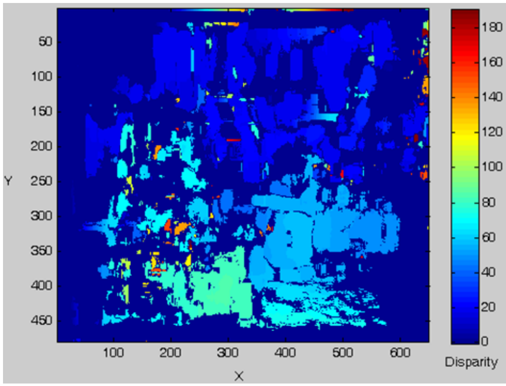

Fig. 7 Graph of computed disparity - SAD

4 FINDING STEREO CORRESPONDENCE USING SAD

The last described method is called SAD - Sum of Absolute Differences. In surroundings of point in left image, quarter little window is defined – in our case the size is 21x21 pixels. Its centre lies in point which coordinate (x coordinate we know, we find only x coordinate) we need find in second image. In the is this little window illustrated. The calculation is based on this equation below:

$$diference(x_p, y) = \sum_{\Delta x=-a}^{a} \sum_{\Delta y=-a}^{a} abs\left[J_1(x_l + \Delta x, y + \Delta y) - J_2(x_p + \Delta x, y + \Delta y)\right] \quad (6)$$

Where *J1* and *J2* are brightness pixel with coordinates (x, y) for left and right images, *a* is half of looking up window - **Fig. 8**.

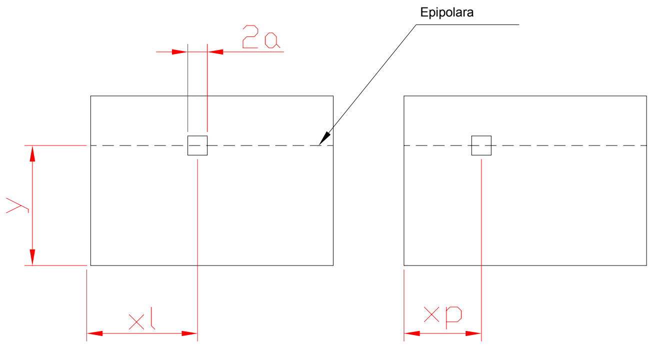

Fig. 8 SAD algorithm

4.1 Result of SAD algorithm

The algorithm was applied on the images, which were tested by Graph Cut algorithm. Computed disparities, which are output of this algorithm, were arranged to graph. The graph shows disparities that belong to pixels in colours according to spectrum on right side. Algorithm doesn't find a correspondence on pixels, which have the zero disparity.

5 CONCLUSIONS

The article shortly described three algorithms suitable for finding matching points in images pairs for stereovision: cross-correlation, graph-cuts and last one SAD - Sum of Absolute Differences.

The cross-correlation is easy and transparent method. Its weak point is finding of not mono-chromatic mask, what can increase the computing time.

The computing time of SAD is much shorter oposit to graph-cut method. On the other hand, we can see that SAD method is not as successfull and robust as graph cuts method.

This article was compiled as part of project FT-TA5/071, supported by the Industrial Research and Development Program of the Ministry of Industry and Trade.

REFERENCES

[1] Gonzales, R. C. *Digital Image Processing*, 3rd, USA, 2008
[2] Kolmogorov, v., Zabih, R. Computing Visual Correspondence with Occlusions via Graph Cuts. In: NY : Cornell University Ithaca, 2001 [cit. 2010-03-31]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.23.3536&rep=rep1&type=pdf
[3] Sojka E., *Digitální zpracování a analýza obrazů* - 1. vyd. - Ostrava : VŠB - Technická univerzita Ostrava, 2000 - 133 s. ISBN 80-7078-746-5
[4] Z. Zhang, *Flexible camera calibration by viewing a plane from unknown orientations.,* Microsoft research technical report 1998, http://research.microsoft.com/˜zhang
[5] D. C. Brown*, Close-range camera calibration,* 1971
[6] Heikilla J., Silvén, O. *A four-step camera calibration procedure with implicit image correction*, 1997
[7] Tsai R. Y., *A versatile camera calibration technique for high accuracy 3D machine vision metrology using off -the-shelf TV cameras and lenses*, 1987
[8] Bouguet, J.Y., *Camera calibration toolbox*, http://www.vision.caltech.edu/bouguetj/calib_doc/index.html
[9] *OpenCV Wiky, Open Source Computer Vision Library Wiki,* http://opencvlibrary.sourceforge.net
[10] Tvarůžka, A. *Senzorický subsystém robotu.* PhD thesis. Department of the Robotics, VŠB-TUO, Ostrava, 139 s., 2008.
[11] J. P. Lewis, *Fast Template Matching*, Vision Interface, p. 120-123, Industrial Light & Magic, USA ,1995.