Application of Computer Vision Technologies for Autonomous Pile Manipulation

Artjoms Suponenkovs
Department of Computer Control Systems
Riga Technical University
Riga, Latvia
Artjoms.Suponenkovs@rtu.lv

AMihails Kovalovs
Department of Image Processing and Computer Graphics
Riga Technical University
Riga, Latvia
mihails.kovalovs@rtu.lv

Zigurds Markovics
Department of Computer Control Systems
Riga Technical University
Riga,Latvia
Zigurds.Markovics@rtu.lv

***Abstract*—Modern robots can perform uncreative monotonous tasks. One of such tasks is pile manipulation. Computer vision technologies can help robots acquire additional information by analyzing a pile of complex objects. One of such complex objects is a fish. The presented work investigates the problems of complex object analysis using computer vision. This paper addresses the challenges of image pre-processing, image segmentation, fish detection and occlusion detection. This work results can be useful for developing a computer vision system for pile manipulation.**

Keywords—computer vision, image pre - processing, fish analysis, pile manipulation, occlusion boundaries, image segmentation.

I. Introduction

Pile manipulation is an exhausting uncreative task. Autonomous industrial robots can do such monotonous tasks. Therefore, the autonomous pile manipulation problem is very relevant. To solve this complicated task, it is reasonable to use computer vision approaches. These approaches allow to analyze visual information on pile of objects.

The autonomous robot can perform various manipulations with objects, depending on the result of visual information analysis [1]. Visual information analysis helps robots to find the necessary object in a pile. In our case, the necessary object is a fish that can be picked up by robot.

A fish can be damaged during displacement and handling. To avoid damaging the fish, robots should pick up only that fish which is not overlapped by any other fish "Fig.1". Therefore, it is important to find the necessary fish without occlusion. To solve this problem, it is possible to use different technologies and approaches: photometric stereo [2], 3D depth sensor [3], the optical flow [4], stereo camera [5], single image [6]. All of these technologies and approaches have advantages and disadvantages. The photometric stereo, for example, is appropriate for tasks that involve simple and big objects. The 3D depth sensor is relatively expensive. It is necessary to analyze a lot of images to realize the optical flow approach. Therefore, in presented work, the information source is a single image. This approach is cheap and fast. For this approach it is not obligatory to have any additional expensive equipment. Also the amount of information is relatively small.

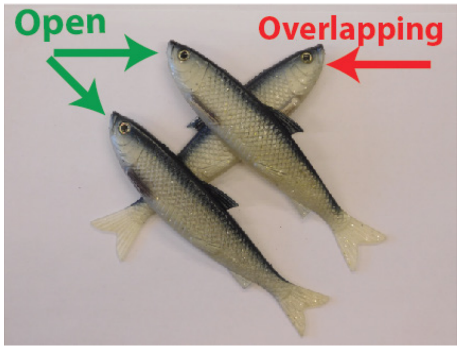

Fig. 1. Open and overlapping fish

II. Materials and methods

The goal of this publication is to find non overlapping fish. To solve this complicated task, it is possible to use the approach that consists of 5 parts "Fig.2":

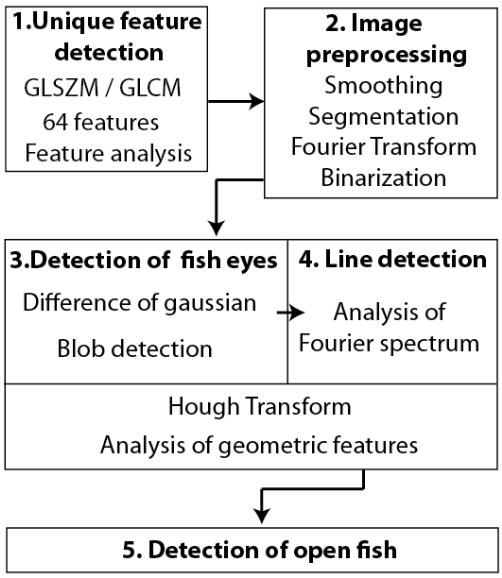

Fig. 2. Fish analysis system flowchart

Print ISSN 1691-5402
Online ISSN 2256-070X
http://dx.doi.org/10.17770/etr2019vol2.4033

A. Unique feature detection

To find open fishes, it is useful to find where fish overlap and fish eye features. It is possible to analyze textural fish features. This analysis can be performed using statistical characteristics, spatial frequencies, structural elements. Gray level matrices allow to calculate statistical characteristics by gray texture data. These statistical characteristics can help to detect fish eye and occlusion. The goal of the statistical characteristic analysis is to find these unique features of fish eye and overlapping.

In this work, we used two types of gray level matrices: Gray Level Co-Occurrence Matrix (GLCM)[7] and Gray Level Size Zone Matrix (GLSZM)[8]. These matrices are used to calculate important statistical features: variance, contrast (1), energy (2), homogeneity(3), correlation, dissimilarity, zone emphasis and etc.

$$CON = \sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2 \tag{1}$$

$$\text{Energy} = \sum_{i,j=0}^{N-1} P_{i,j}^{\ 2} \tag{2}$$

$$\text{Homogeneity} = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+|i-j|} \tag{3}$$

where:

P – is the probability of combined neighboring elements (GLCM value);
i and j – are GLCM indexes;
N – size of GLCM matrix.

It is possible to design a computer vision system for the analysis of open and overlapping fish based on detected features in the first experiment (looking below).

B. Image preprocessing (image preparation)

At this stage, it is important to make an input image preparation for further analysis. The image preparation depends on the type of further analysis. At next stages, it is possible to use different types of analysis: analysis of geometric features, analysis of Fourier spectrum and analysis of textural features. Therefore, it reasonable to use the Fourier transform together with the Fourier spectrum analysis, segmentation together with the analysis of geometric features, difference of Gaussian together with analysis of textural features.

Usually the input image contains noise. This noise makes it difficult to analyze the fish. At this stage, it is very important to reduce the noise level by using special smoothing algorithms. It is reasonable to use Perona-Malik filtering [9]-[10], Gaussian filtering or Mean Shift filtering to reduce noise. Perona-Malik method has a useful feature that saves strong edges of objects in the image. The "Fig. 3" shows the results of Perona - Malalik filtering, Mean Shift filtering and K-Means clustering [11]-[12]. Perona - Malalik and Mean Shift filtering "Fig. 3 B and C" remove high frequency information, which in turn is helpful for the segmentation of the input image.

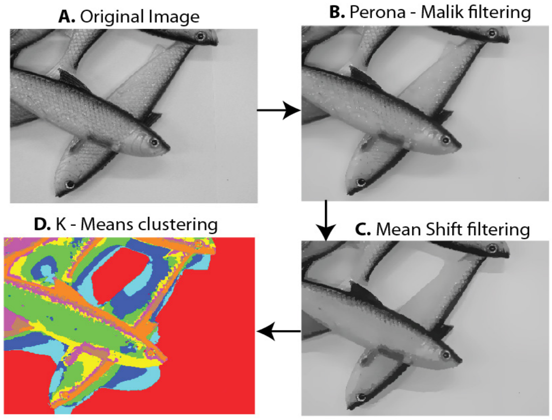

Fig. 3. Input image smoothing and clustering

The segmentation of input image is important for the fish analysis. The input image can be divided into many pixel regions by performing image segmentation. After that it is possible to analyze each region by using geometrical features.

The "Fig. 4" shows the steps of segmentation. The first step "Fig. 4 B" is K-Means clustering. K-Means clustering divides an image into 8 clusters. This division is based on pixel intensity. The second step "Fig. 4 C" is binarization of regions of interest (ROI). This binarization is based on cluster features. The third step "Fig. 4 D" is segmentation of separate regions.

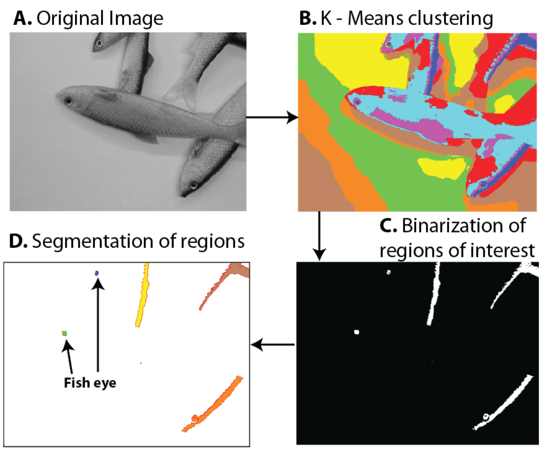

Fig. 4. Steps of segmentation

C. Detection of fish eyes

The eye is the most easily perceptible part of a fish. Therefore it is reasonable to detect the fish eye at first. Detection of the fish eye is based on a unique feature detection experiment (see below). There are many unique features. For example variance, contrast, dissimilarity. By using these features we can analyze the image texture and check if it contains the fish eye.

In this work we are trying to use many methods of fish eye detection: the Difference of Gaussian (DoG), blob detection [13], analysis of geometric feature, Hough circle detection. All these methods have advantages and disadvantages.

Blob detection is based on the Laplacian of Gaussian

(LoG) [14]. Our experiment (see below) shows that a fish eye has a high Laplacian response. That would be because the fish eye is an ideal blob and the fish eye looks like as Laplacian filter. It is possible to simplify Laplacian calculation by Difference of Gaussian (DoG). The DoG has approximately the same result as scale-normalized Laplacian. Scale-normalized Laplacian and DoG can be calculated as follows (4),(5),(6):

a) Gaussian:

$$G(x,y,\sigma)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{(-\frac{x^2+y^2}{2*\sigma^2})} \tag{4}$$

b) Difference of Gaussian:

$$DoG = G(x,y,k\sigma)-G(x,y,\sigma) \tag{5}$$

c) Scale-normalized Laplacian:

$$LoG=(G_{xx}(x,y,k\sigma)-G_{yy}(x,y,k\sigma)) \tag{6}$$

where

x, y are pixel coordinates;

σ - deviation;

k - coefficient;

$G_{xx,}$ G_{yy} - second partial derivatives.

It is possible to analyze segment shape by its geometric features. The fish eye is oval or a circle. Therefore, we can use the circularity. The circularity of a circle is (4 * π), hexagon - 13.86, square - 16 and of equilateral triangle - 20.79. The circularity can be calculated as follows (7):

$$C=\frac{P^2}{S} \tag{7}$$

where:

P - is segment perimeter,

S - is segment area.

After calculation of circularity of a segment, we can make comparison of the calculated circularity and known circularity of ideal shapes (circle, hexagon, square, circle). The other important geometric feature is elongation (8).

$$Elong=\frac{m_{20}+m_{02}+\sqrt{(m_{20}-m_{02})^2+4*m_{11}{}^2}}{m_{20}+m_{02}-\sqrt{(m_{20}-m_{02})^2+4*m_{11}{}^2}} \tag{8}$$

where:

m_{20} - is two-dimensional central moment (2 and 0 is moment index);

m_{jk} - is central moment (j and k is moment index).

These geometric features are useful for circle detection.

D. Line detection

The Line detection can be performed using Hough transformation [15]-[16] that generates a list of lines. Long lines are more important than short. If a fish has relatively long line, then this fish is probably not overlapped by other fish “Fig. 5“. The “Fig. 5“ shows the relation between length of line and fish overlapping. This relation points to the importance of long lines.

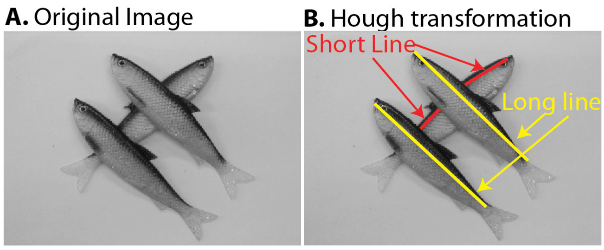

Fig. 5. Line detection

Additional information about fish orientation can be helpful for further analysis. This information about orientation allows us to remove unnecessary fish lines and help detect an open fish “Fig. 7“. The fish orientation can be determined by spectral analysis. It is for this reason that the original image must be converted into spectral image by performing Fourier transform. The “Fig. 6“ shows the determination of fish orientation. It is possible to estimate the spectral image “Fig. 6 B“ by using the following equation(9):

$$F(\theta)=\sum_{r=1}^{R_{\max}}F_r(\theta) \tag{9}$$

where

$F_r(\theta)$ - is a function (polar coordinate system) that returns spectral image pixel intensity;

$F(\theta)$ - is function that returns a sum of pixel intensities;

θ - a corner;

r - a radius;

R_{max} - the maximum radius.

This equation(9) allows us to find the corner θ_{max} which matches the maximum sum of pixel intensities (F_{max} (θ_{max}) = maximum). It is possible to use a threshold for the detection of the maximum sum of pixel intensities. In that case, we can find many important corners (θ_{max1}, θ_{max2}). For example, the spectral image has two important corners (in the “Fig. 6 B and C“). The angles of these two corners must be increased by 90 degrees ($\varphi=\theta_{max}+90°$). This creates two new corners (φ_1, φ_2) that indicate the fish orientation (see “Fig. 6 A and D“).

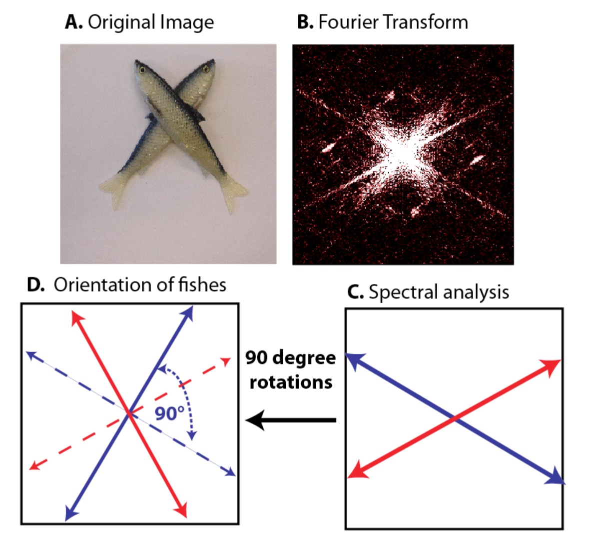

Fig. 6. Orientation of fishes

E. Detection of an open fish

The detection of an open fish has 3 steps "Fig. 7":

1) detection of the fish eye;

2) finding long lines around the eye;

3) checking the orientation of the line.

As a result of this detection there are fish that are probably not overlapped by other fish.

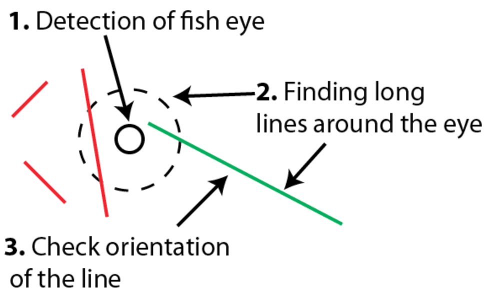

Fig. 7. Detection of an open fish

III. Results and Discussion

This part contains the results of three experiments:

1. Unique feature detection;
2. Detection of fish eyes;
3. Detection of fish lines.

A. Experiment - Unique feature detection

This experiment investigates the problem of unique feature detection. Unique features were detected using the GLCM and GLZM matrices. These matrices are used to analyze textural information. Sixty-four textural features were taken into account in this experiment. For this purpose, one GLZM matrix and four GLCM matrices (horizontal, vertical and two diagonal) were used. The GLZM matrix describes 16 statistical features and one GLCM matrix describes 12 statistical features. As a result there are 64 features (16 + 12 * 4).

TABLE I. Feature comparison between fish overlapping and background texture

Feature number/s	Feature/s	is greater for	Confidence
GLCM(20, 21, 44, 45, 8, 9, 32, 33)	Mean X and Mean Y	Background	0.903
GLZM (51 - LGZE)	Low Gray level Zone Emphasis	Fish overlapping	0.901
GLZM(60 - BARYGL)	The barycenter on gray level	Background	0.895
GLZM(52 - HGZE)	High Gray level Zone Emphasis	Background	0.883
GLCM(22, 23, 46, 47, 34, 35, 10, 11)	Variance X and Variance Y	Fish overlapping	0.839

The table 1 shows the important features. This table results are based on 342 comparisons between overlapping fish and background texture. As a result of the comparison there are some important features: mean, low gray level zone emphasis, barycenter, high gray level zone emphasis and variance. These features are helpful in the detection of overlapping fish. The table also contains the confidence level that shows the importance of a given feature.

The table 2 shows the important features of fish eye texture. This table results are based on 350 comparison between fish eye and background texture. As a result of comparison there are some important features: variance, correlation, contrast, large zone high gray level emphasis and dissimilarity.

The detection of fish eyes and the detection of fish lines that are described above, are based on the results of this experiment.

TABLE II. Feature comparison between fish eye and the background

Feature number/s	Feature/s	is greater for	Confidence
GLCM (22, 23, 46, 47, 34, 35, 10, 11)	Variance X and Variance Y	Fish eye	0.992
GLCM (36, 24, 12, 48)	Correlation	Background	0.991
GLCM (37, 1, 25, 13)	Contrast	Fish eye	0.989
GLZM (LZHGE - 56)	Large Zone High Gray level Emphasis	Background	0.986
GLCM (26, 2, 38, 13, 14)	Dissimilarity	Fish eye	0.984

Experiment - Detection of fish eyes

This experiment investigates the problem of the detection of fish eyes. The detection of fish eyes was performed using 5 methods:

1) BLOB - the blob detection;
2) DOG - method is based on the Difference of Gaussian;
3) DOG + BLOB - combination of 1st and 2nd methods;
4) SEGMENT - method is based on segmentation and geometric feature analysis.
5) HOUGH - the Hough transformation (circle detection).

The "Fig. 8" and "Fig. 9" show comparison between results of these methods. There are 6 bars in "Fig. 8". The "Original" bar shows the number of fish eyes in original image. As shown in "Fig. 8" there are two methods that have good results: DOG and DOG + BLOB. The DOG method had detected 86 percent of fish eyes, but DOG + BLOB had detected 90 percent of fish eyes.

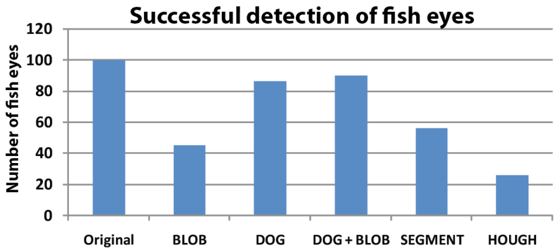

Fig. 8. Number of successful detected eyes

The "Fig. 9" shows the number of mistakes of each method. As shown in "Fig. 9" BLOB has the least number of mistakes. Then that means BLOB is very stable method. Therefore it is possible to make a combination of BLOB and other methods (for example: BLOB + DOG).

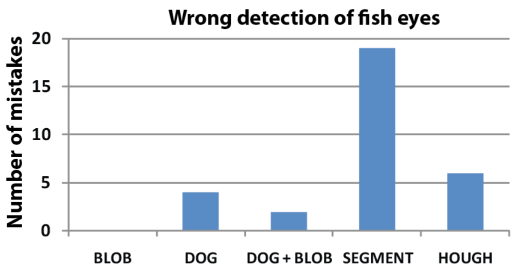

Fig. 9. Number of incorrectly detected eyes

Experiment - Detection of fish lines

This experiment investigates the problem of the detection of fish long lines. The detection of fish lines was performed using 2 methods:

1) HOUGH - the Hough transformation (line detection);
2) SMART HOUGH - the Hough transformation and texture analysis.

The "Fig. 10" shows comparison between the results of these methods. As shown in "Fig. 10" SMART HOUGH has the lowest number of mistakes because of texture analysis that removes unnecessary lines.

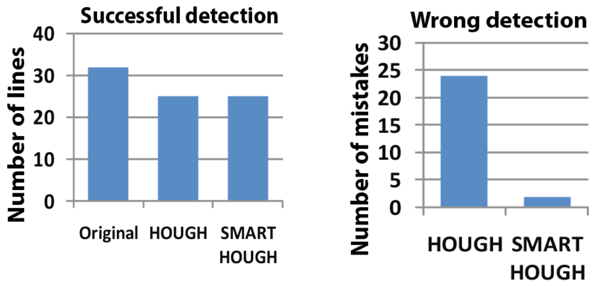

Fig. 10. Number of wrong and successful detected fish lines

IV. Conclusions

The results of the first experiment show that textures of overlapping fish and fish eye have unique statistical features. The unique statistical features of overlapping fish are mean value, low gray level zone emphasis, the barycenter on gray level, etc. The unique statistical features of fish eye are variance, correlation, contrast, etc. It is possible to design computer vision system for the analysis of open and overlapped fish based on detected unique features.

The results of the second experiment show that DOG + BLOB had detected 90 percent of fish eyes. The comparison of fish eye detection methods show that BLOB method has the lowest number of mistakes. Therefore it is possible to make a combination of BLOB and other methods. The combination of BLOB and DOG methods has the best results.

The results of the third experiment show that SMART HOUGH had detected approximately 78 percent of fish lines. SMART HOUGH has the lowest number of mistakes because of texture analysis that removes unnecessary lines.

This work results can be useful for developing a computer vision system for pile manipulation.

Acknowledgments

The authors wish to acknowledge support from PERUZA company of Latvia "Fig. 11".

Fig. 11. Logotype of PERUZA

References

[1] R.Mojtahedzaden, Safe Robotic Manipulation to Extract Objects from Piles: From 3D Perception to Object Selection, Orebro University, Repro, 2016.

[2] Ikeuchi K., B. K.P. Horn, S. Nagata, T. Callahan and O. Feirigold, "Picking up an object from a pile of objects". Massachusetts Institute of Technology, 1983, A.I. Memo, 726: 1-26.

[3] Domae Y., Okuda H., Taguchi Y., Sumi K. and Hirai T., "Fast Graspability Evaluation on Single Depth Maps for Bin Picking with General Grippers", IEEE International Conference on Robotics and Automation, 2014.

[4] Patrik Sundberg, Thomas Brox, Michael Maire, Pablo Arbelaez and Jitendra Malik, "Occlusion Boundary Detection and Figure/Ground Assignment from Optical Flow", IEEE, 2011.

[5] Wang X., H. Xiong and L. Li, "Stereo Vision Based Piled Object Recognition System for Robot", Ricoh Software Research Center Co., Ltd., 41: 31-36, 2016.

[6] Derek Hoiem, Andrew N. Stein, Alexei A. Efros and Martial Hebert, "Recovering Occlusion Boundaries from a Single Image ", Int J Comput Vis 91: 328–346, 2007.

[7] Fritz Albregtsen, "Statistical Texture Measures Computed from Gray Level Coocurrence Matrices", University of Oslo, 2008.

[8] Guillaume Thibault, Jesus Angulo, Fernand Meyer, "Advanced statistical matrices for texture characterization: application to dna chromatin and microtubule network classification", 18th IEEE International Conference on Image Processing, 2011.

[9] Joachim Weickert, "Anisotropic Diffusion in Image Processing", Department of Computer Science University of Copenhagen, Copenhagen, Denmark, 2008.

[10] Maciej Wielgus, "Perona-Malik equation and its numerical properties", Lipiec, 2010.

[11] Aristidis Likas, Nikos Vlassis, Jakob J. Verbeek, "The global k-means clustering algorithm", Elsevier Pattern Recognition 36 (2003) 451 – 461.

[12] Hans-Hermann Bock, "Origins and extensions of the k-means algorithm in cluster analysis", Electronic Journal for History of Probability and Statistics, Vol 4, December 2008.

[13] Anlong Ming, Huadong Ma, "A blob detector in color images", CIVR'07, 2007, Amsterdam, The Netherlands.

[14] Hui Kong, Hatice Cinar Akakin, and Sanjay E. Sarma, "A Generalized Laplacian of Gaussian Filter for Blob Detection", IEEE Transactions on cybernetics, vol. 43, no. 6, December 2013.

[15] Varun R., Yadunandan Vivekanand Kini, K. Manikantana and S. Ramachandranb, "Face Recognition using Hough Transform based Feature Extraction", ICICT, Procedia Computer Science 46, 1491 – 1500, 2015.

[16] Antoine Manzanera, Thanh Nguyen, Xiaolei Xu, "Line and circle detection using dense one-to-one Hough transforms on greyscale images", EURASIP Journal on Image and Video Processing, Springer, 2016, 34, pp.1773 - 1773.