Real-time centre detection of an OLED structure

Real-Time Center Detection of an OLED Structure

Roel Pieters, Pieter Jonker, and Henk Nijmeijer

Dynamics and Control Group, Department of Mechanical Engineering
Eindhoven University of Technology, PO Box 513, 5600 MB Eindhoven
{r.s.pieters,p.p.jonker,h.nijmeijer}@tue.nl

Abstract. The research presented in this paper focuses on real-time image processing for visual servoing, i.e. the positioning of a x-y table by using a camera only instead of encoders. A camera image stream plus real-time image processing determines the position in the next iteration of the table controller. With a frame rate of 1000 fps, a maximum processing time of only 1 millisecond is allowed for each image of 80x80 pixels. This visual servoing task is performed on an OLED (Organic Light Emitting Diode) substrate that can be found in displays, with a typical size of 100 by 200 μm. The presented algorithm detects the center of an OLED well with sub-pixel accuracy (1 pixel equals 4 μm, sub-pixel accuracy reliable up to ± 1 μm) and a computation time less than 1 millisecond.

1 Introduction

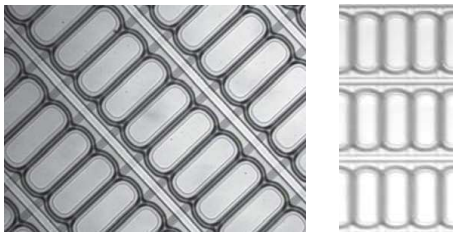
Present day measurement and positioning systems are pushing the limits regarding positioning accuracy and fabrication time. For this, visual servoing is a method that broadens the area for intelligent embedded vision systems (see for instance [3], [5] and [2]). In literature, control in visual servoing is a research topic characterized by various classifications and formats. An extensive review can be found in [5]. Image processing algorithms for this are described e.g. in [8], while vision systems that use image processing to obtain 1 ms visual feedback with a massively parallel architecture are presented in [10] and [6]. However, the combination of image processing and high performance visual servoing in real-time, *without* massively parallel processing is rather new. This combination, with an emphasis on real-time image processing embed in smart cameras, is presented in this paper.

In visual servoing, the measurement loop between product and sensor is cut short by placing a camera directly on the production head. This means that the measuring system now no longer relies on encoders for positioning but solely on the camera. The most limiting factor in this is the balance between a high sampling rate and the size of the image to extract useful information. An equilibrium is found for our OLED application characterized by positioning at micrometer scale and a frame rate of 1 kHz (1000 fps) by using an ROI of 80×80 pixels, whereas 1 pixel equals 4 μm. The product to be controlled is in our case a substrate (see figure 1), consisting of a 2D repetitive pattern of OLED 'wells' or

J. Blanc-Talon et al. (Eds.): ACIVS 2009, LNCS 5807, pp. 400–409, 2009.

'cups' which need to be filled (printed) with a polymer substance by an industrial inkjet printer. Each OLED has a typical size of 100×200 μm that should be filled by a slightly larger droplet with an accuracy of maximum 10 μm. As the measurements are taken at 1 kHz, this results in a maximum effective computation time of 1 ms. The captured 80×80 pixels then contain 3 OLED structures. For visualization purposes, this paper deals with images of 100×200 pixels and thus nine OLED cups. The method we present here is split into an on-line step and an off-line step. The off-line step does an initial calibration of the substrate to ensure a margin of alignment with respect to the camera. If necessary, this can be accompanied by a shading correction if the grey-level uniformity is out of bound, and a height adjustment of the camera if the image is out of focus. The on-line step consists of the actual movement of the camera over the substrate from which images are taken at a fixed rate of 1 kHz. The OLED cups are then extracted with a Difference of Gaussian filter which separates the cups from their surroundings. Subsequent morphological operations Erosion and Dilation remove noise. From the resulting blobs the outlines are drawn by following the contours and an area calculation leaves only the nine largest structures. A larger rectangular box is drawn around each structure and used as an ROI (region of interest) on the original image to determine the center of gravity, which gives the center of each cup. Also on-line, a measure of focus can be calculated, as well as an extra orientation calculation with the resulting found cup centers. This paper is organized as follows: Section 2 describes the algorithms used in the off-line step in detail. The algorithms used in the on-line step are explained in Section 3. Results of the performed experiments are given in Section 4. Finally, conclusions and future work are given in Section 5.

The long-term goal of our project 'Fast Focus On Structures' (FFOS) is to develop an intelligent visual servoing system that can be used for various industrial applications such as industrial inkjet printing. This paper mainly focuses on the embedded image processing task for visual servoing, being the extraction of highly accurate positions and orientations (poses) for the control of an x-y table. The motion control side of this project can be found in [2].



(a) OLED substrate (b) image

Fig. 1. OLED substrate. High resolution image of OLED substrate (a). The OLED images are rendered in grey scale with, in our case, nine OLEDs per image. The size of this image is 100×200 pixels (b). Image (a) taken from OTB Group.

2 Off-Line Step

The off-line step is used for the calibration of the camera with regard to the OLED substrate. The start point on the substrate for the on-line step should be found and some calibration criteria should be checked. A single image is used to determine if a correction in alignment, depth of focus and shading is necessary. Since in this phase timing is not essential, the image can be as large as the camera allows. This check can be done at several locations on the substrate and with several image sizes.

2.1 Histogram of Gradient Orientations

The orientation of the complete OLED substrate / complete display should be determined with an accuracy of $\pm 1\,^{\circ}$. This is done to ensure a roughly correct alignment with respect to the initial camera frame. The camera orientation angle can only be corrected up to 5 degrees in the on-line stage, as the center detection algorithm in the on-line stage is only guaranteed between [-5:5] degrees (see section 3). The orientation of the entire OLED is determined by calculating the histogram of gradient orientations from the OLED image, which is convoluted with two different 3 x 3 Sobel kernels:

$$S_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \qquad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{1}$$

These resulting gradient approximations are then used to calculate the gradient direction in every pixel:

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right) \tag{2}$$

where G_x and G_y is the result of the image convoluted with S_x and S_y respectively. This gradient calculation has an accuracy of $\pm 1\,^{\circ}$, due to the inaccurate

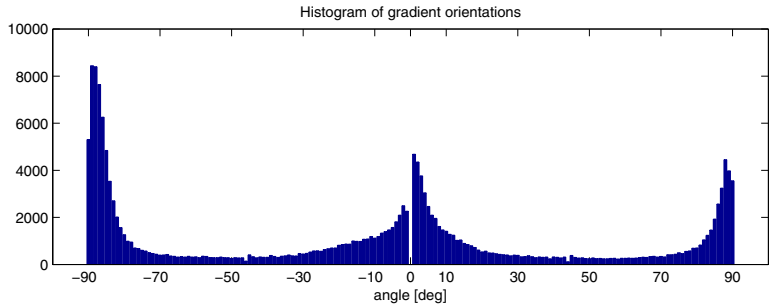


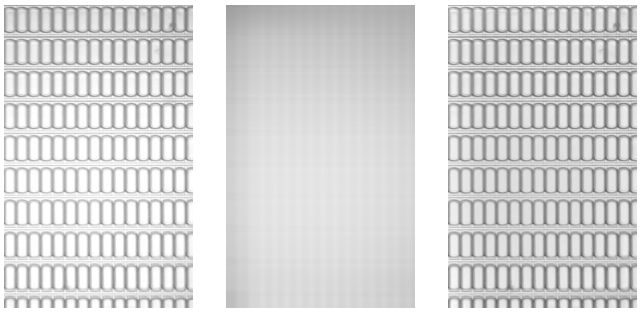
Fig. 2. Histogram of gradient orientations. Image gradients are calculated using 3x3 Sobel kernels. The two peaks represent the vertical (biggest peak) and horizontal edge of the OLED grid structure.

approximation of the image gradient. From the gradient directions a histogram is constructed which depicts peaks in horizontal (x) and vertical (y) directions (see figure 2). These peaks depict the horizontal and vertical edges of the OLED grid.

When the resulting gradient exceeds a threshold (i.e. $|\Theta| > 5\,^\circ$), the camera orientation should be adjusted or the OLED structure should be repositioned.

2.2 Shading Correction

Depending on the choice and design of the final vision system (i.e. camera, lens, lighting, etc) a *shading correction* can be done off-line and on-line. On-line correction is possible due to its relatively low computation time. The shading is estimated e.g. by morphological filtering [13], where a smoothed version (Gaussian) of the input image is subtracted from the original input image. This smoothed version is the estimate of the background (see figure 3). The standard deviation (sigma) of the 9×9 Gaussian smoothing kernel is determined from the standard deviation of the input image.



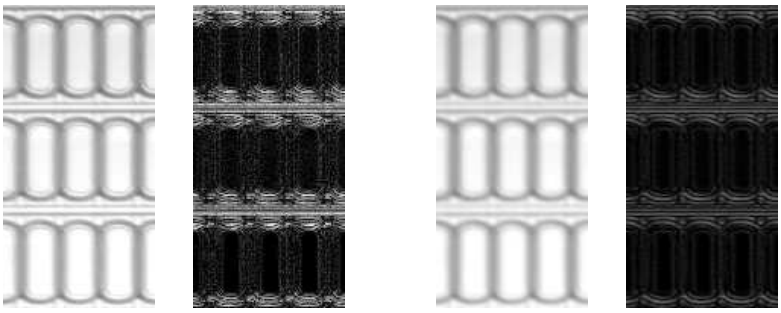
(a) input (b) smoothed (c) corrected

Fig. 3. Shading correction. Figure (a) shows the original input image. Figure (b) shows the smoothed version of (a). Figure (c) shows the image corrected for shading.

2.3 Depth of Focus

A sharp focus on the OLED substrate is needed to ensure a reliable pose calculation for each cup. For this a sharpness of focus measure is used. As an image with sharp focus has more high-frequency content than a blurry and badly focused image, a Laplacian operator is used that responds to high-frequency variations of image intensity and produces maximum values when the image is perfectly in focus. Since the terms in the Laplacian can have opposite signs which can cancel each other, a modified Laplacian is used [11]:

$$\nabla^2_m I = \left|\frac{\partial^2 I}{\partial x^2}\right| + \left|\frac{\partial^2 I}{\partial y^2}\right| \tag{3}$$



(a) sharp (b) laplace (c) blurred (d) laplace

Fig. 4. Depth of focus. Figure (a) and (b) show the sharp image with its Laplacian. The sum of the sharp Laplacian is 0.3813. Figure (c) and (d) show a blurred image (sharp image convoluted with 5 x 5 Gaussian) with its Laplacian. The sum of the non-sharp Laplacian is 0.1903, which is significantly less. Note that even with the blurred image still all OLED centers are found.

with $I = I(x, y)$ an input image. The normalized measure of focus is then defined as the normalized sum of all values in the modified Laplacian:

$$F_{Mn} = \sum_{i=0}^{n} \sum_{j=0}^{m} \nabla_M^2 I * \frac{1}{128mn} \tag{4}$$

with n and m the width and height of an image and 128 the normalization factor. When this sum deviates too much from a given threshold (i.e. $F_{Mn} < T_M$), then the height of the camera should be adjusted. This could be done manually or by means of a routine which searches for the highest F_{Mn}. The measure of focus can be calculated off-line but even on-line due to its relatively low computation time. Extensive experimentations with various substrates have to indicate whether an on-line implementation is necessary.

3 On-Line Step

In the on-line step computation time is an important issue. The available single millisecond should be used as effectively as possible. However, some margin should be left to correctly close the computations and to cope with possible delay. To reach the required 1 kHz sampling rate the largest possible image is 80×80 pixels. This contains three OLED cups horizontally separated. To visualize the operations, for this paper we used an image of 100×200 pixels, i.e. nine OLED cups.

3.1 Difference of Gaussians

The images are processed in scale space; an image is represented as a single-parameter family of smoothed images, parameterized by the size of the smoothing kernel, i.e. a low pass filter. It is defined as the function, $L(x, y, \sigma)$, which

is produced by convolving a variable-scale Gaussian, $G_\sigma(x, y)$, with an input image, $I(x, y)$:

$$L(x, y, \sigma) = G_\sigma(x, y) * I(x, y), \tag{5}$$

where $*$ represents the convolution in x and y, and

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{6}$$

with σ the variance (width) of the Gaussian kernel.

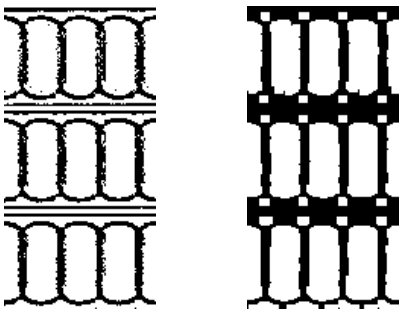
Blobs can now be found using scale-space extrema in the Difference-of-Gaussian function ($D(x, y, \sigma)$) convoluted with the input image:

$$\begin{aligned} D(x, y, k, \sigma) &= (G_{k\sigma}(x, y) - G_\sigma(x, y)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{7}$$

where the difference of two nearby scales is separated by a constant multiplicative factor k [9], [7]. This is equivalent to a band-pass filter that preserves frequencies that lie in the range of both images.

This function is computationally simple - only convolution and subtraction - and the pixel accuracy remains the same before and after computation, i.e. there is no scaling down. After blob-detection, the image is binarized with blobs equal to '1' and the background equal to '0' (see figure 5a).

As can be seen in figure 5a numerous blobs are detected, where some blobs are artifacts of neighboring OLEDs or line segments of the substrate. Also noise can be seen at either side of the OLEDs, which potentially could disturb future computations, such as the contour following. To remove this noise and most of the line segments, the binary image is eroded and subsequently dilated with a 3×3 square structuring element.

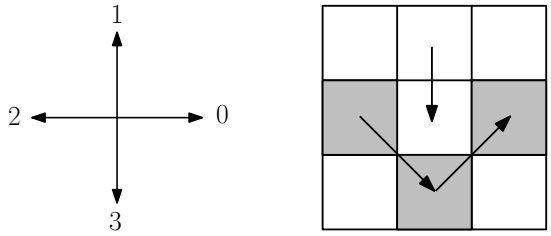


(a) DOG (b) noise removal

Fig. 5. Subtraction of structural element. Figure (a) shows the result after blob-detection (Difference of Gaussians). Figure (b) shows the result after twice erosion and dilation for removal of noise and line segment artifacts.

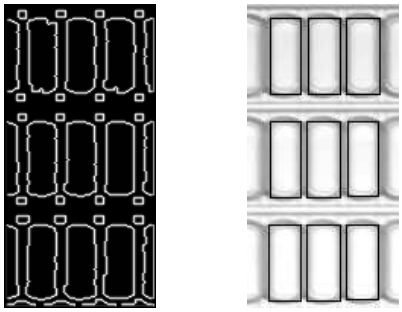
3.2 Boundary Following

After blob-detection and noise removal the OLED cups and some line segment artifacts (figure 5b) are left. We now follow the boundaries of each binary blob. From the upper left pixel in the binary image a row-wise search is made to find the first '1'. When found, this contour is followed in a counter-clockwise manner (see figure 6b), where each contour point is stored into memory as a Freeman chain code (FCC) [4]. FCC is a compact way to represent a contour of an object. It is basically a sequence of directions of the steps taken when following the boundary of a contour (see figure 6a). Each blob is described in this way until the end of the image is reached (figure 7a).



(a) Direction (b) pixel search

Fig. 6. Boundary following. Figure (a) shows the direction notation for 4-connectivity detection. Figure (b) shows the pixel neighborhood search for 4-connectivity (counter-clockwise) starting from direction '3' (arrow down).



(a) Boundary following (b) 9 biggest contours

Fig. 7. Figure (a) shows the result after boundary following. From this binary image, the nine biggest contours are kept. Figure (b) shows the rectangles (ROIs) which are drawn around each of the nine contours and are used for center of gravity calculation in the input image.

3.3 Center of Gravity

The area of each contour is then calculated and only the nine largest contours are kept which represent the nine OLED cups (figure 7b). A slightly larger ROI

is then set around each contour and used as an ROI in the *original* input image to finally calculate the center of gravity. Since the OLED structure is symmetric and we assume there are no significant lighting variations over the OLED, this is also the *center* of the OLED (figure 8a).

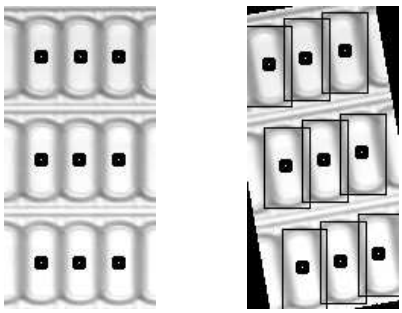
4 Experiments and Results

Since the lighting over one OLED (100 by 200 μm) does not change significantly, the algorithm is invariant to lighting or shading differences. The detection of the blobs is invariant to orientation, however, the calculation of the center is not. After the contours are found a rectangle is drawn around it. When the substrate is rotated slightly ($|\alpha_I| > 10\,^\circ$, with α_I the rotation angle of the OLED substrate), the rectangle does not cover the OLED cup completely anymore (see figure 8b). For this, a lower limit is set in orientation error by correcting a false offset in the off-line step. A more accurate orientation can be calculated in the on-line step by simple triangulation. Each maximum and minimum center point for each row and column on a 3×3 OLED grid is known and can be used to calculate the orientation relatively accurate ($\pm 0.4\,^\circ$):

$$\tan \alpha_I = \frac{\Delta y}{\Delta x}. \tag{8}$$

with Δ the difference between successive points in x- and y-direction. This can then be used to correct a rotational offset.

Since the center of gravity is a calculation not rounded off to integers, the accuracy of the complete algorithm is sub-pixel, with each pixel being 4 μm in height and width. However, since the precise accuracy is difficult to determine (see e.g. [1]), and the actuation of the x-y table could be limited to a certain precision, the sub-pixel accuracy is said to be reliable up to ± 1 μm. The accuracy specification (maximum 10 μm) has therefore been largely met. Extensive experiments with different substrates under various conditions (i.e. lighting) should give a better proof of performance.



(a) Centers of OLEDs (b) incorrect ROI overlay

Fig. 8. Result of algorithm. Figure (a) shows the centers of the nine OLED cups. Figure (b) shows an incorrect ROI overlay when the substrate exceeds an orientation of $10\,^\circ$.

The on-line algorithm is controlled by the timer of the camera, which is set to stream images at a fixed rate (1 [kHz]). Whenever a frame is done and send to the computer, a callback ensures immediate processing on the obtained image. The computation time of the entire on-line algorithm is less then 2 milliseconds, hence each OLED center detection takes about 2/9 milliseconds. The experiments have been carried out under Real-time Linux Ubuntu 8.10 in combination with C/C++ code on a standard notebook with 2 GB of RAM and a 2.4 GHz Intel Core 2 Duo CPU.

5 Conclusions and Future Work

The presented real-time center detection algorithm shows to be a suitable candidate for the computation of center coordinates of an OLED substrate in a visual servoing framework. The algorithm is split in an off-line step for calibration and an on-line step for real-time image processing. Off-line, orientation alignment is checked using a histogram of gradient orientations and a measure of focus is calculated using the Laplacian operator. On-line, the algorithm combines blob-detection, boundary following and center-of-gravity calculation to obtain sub-pixel accuracy (1 pixel equals 4 μm, sub-pixel accuracy reliable up to ± 1 μm).

With an image of 80 x 80 pixels the computation time is less than 1 millisecond per image, making 1 kHz visual servoing possible. Since the algorithm is controlled with the timer of the camera, the main source of delay is due to image processing. The variation of this delay is highly platform dependent. Therefore, further research should be carried out to gain more knowledge in this. Also for future research, an industrial variant will be implemented on an FPGA to cope with delay and to obtain even faster computation. Testing on an actual (moving) setup will provide insight into more complex effects such as delay, lighting influences, abnormalities in alignment, surface inequality, vibrations, etcetera. These effects and their possible solutions can then be subjected to a more close investigation. This will also give insight in the necessity for on-line shading correction and depth of focus adjustment as well as a more precise positioning accuracy.

Acknowledgement

This research was supported by SenterNovem - IOP Precision Technology - Fast Focus On Structures (FFOS).

References

1. van Assen, H.C., Egmont-Petersen, M., Reiber, J.H.C.: Accurate Object Localization in Gray Level Images using the Center of Gravity Measure; Accuracy versus Precision. IEEE Transactions on Image Processing 11(12), 1379–1384 (2002)
2. de Best, J.J.T.H., van de Molengraft, M.J.G., Steinbuch, M.: Direct Dynamic Visual Servoing at 1 kHz by Using the the Product as One Dimensional Encoder. In: 7th IEEE International Conference on control and Automation, New Zealand (submitted 2009)

3. Czajewski, W., Staniak, M.: Real-time Image Segmentation for Visual Servoing. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. LNCS, vol. 4432, pp. 633–640. Springer, Heidelberg (2007)
4. Freeman, H.: Computer processing of line-drawing images. Computing Surveys 6(1), 57–97 (1974)
5. Hutchinson, S., Hager, G.D., Corke, P.I.: A tutorial on visual servo control. IEEE Trans. on Robotics and Automation 12(5), 651–670 (1996)
6. Ishii, I., Nakabo, Y., Ishikawa, M.: Target tracking algorithm for 1ms visual feedback system using massively parallel processing. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2309–2314 (1996)
7. Kanters, F.M.W.: Towards Object-based Image Editing. Phd thesis, Eindhoven University of Technology (2007)
8. Loncaric, S.: A Survey of Shape Analysis Techniques. Pattern Recognition 31(8), 983–1001 (1998)
9. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
10. Nakabo, Y., Ishikawa, M., Toyoda, H., Mizuno, S.: 1 ms Column Parallel Vision System and It's Application of High Speed Target Tracking Robotics and Automation. In: Proc. IEEE Int. Conf., vol. 1, pp. 650–655 (2000)
11. Riaz, M., Park, S., Ahmad, M.B., Rasheed, W., Park, J.: Generalized Laplacian as Focus Measure. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part I. LNCS, vol. 5101, pp. 1013–1021. Springer, Heidelberg (2008)
12. Rodriguez, J., Ayala, D.: Erosion and Dilation on 2D and 3D Digital Images: A new size-independent approach. In: 6th International Fall Workshop on Vision, Modeling and Visualization, pp. 143–150 (2001)
13. Young, I.T.: Shading Correction: Compensation for Illumination and Sensor Inhomogeneities. In: Robinson, J.P., et al. (eds.) Current Protocols in Cytometry, pp. 2.11.1–2.11.12. John Wiley and Sons, Inc., Chichester (2000)