

# Automatic Turbot Fish Cutting Using Machine Vision

Fernando Martín-Rodríguez<sup>(1)</sup>, Mónica Barral-Martínez<sup>(1)</sup>.

<sup>(1)</sup> Departamento de Teoría de la Señal y Comunicaciones. Universidad de Vigo.  
E.E.T. C/ Maxwell S/N. 36310 Vigo (Pontevedra).

Phone: +34-986-812151, Fax: +34-986-812116, E-mail: [fmartin@tsc.uvigo.es](mailto:fmartin@tsc.uvigo.es).

**Abstract-** *This paper is about the design of an automated machine to cut turbot fish specimens. Machine vision is a key part of this project as it is used to compute a cutting curve for specimen's head. This task is impossible to be carried out by mechanical means. Machine vision is used to detect head boundary and a robot is used to cut the head. Binarization and mathematical morphology are used to detect fish boundary and this boundary is subsequently analyzed (Hough transform) to detect key points and thus defining the cutting curve. Afterwards mechanical systems are used to slice fish to get an easy presentation for end consumer (as fish fillets than can be easily marketed and consumed)*

**Keywords:** *turbot fish, food industry, machine vision, mathematical morphology, Hough transform.*

## I. INTRODUCTION

There is some interest in food industry to be able to automatically slice turbot fishes to get a new commercial presentation. It would consist of fish fillets with no fish bones, very easy to cook and to eat, probably very attractive to customers. Turbots are flat fishes; they have their two eyes on the same side as they always swim on the bottom. They also have a very delicate flavor and high commercial interest. Nowadays, they are farmed in numerous places, so that it is easy and cheap getting big amounts of individuals with great size variety.

Its special shape makes difficult to create an automatic cutter. The main problem is cutting the fish head, as it is necessary to develop a curve cutting that will depend on specimen size. Formerly this cutting was made by specialized personnel but it is an unpleasant and dangerous activity because specimens are very slippery and can cause that workers hurt themselves with their own knives (very big knives are used for this).

That's the reason for using computer vision as a means of automatically detecting the necessary cutting curve and instructing a robot for cutting the head away (a XML file is used to communicate the curve to an industrial robot). Afterwards, purely mechanical systems are used to slice fish to get an easy presentation for end consumer.

## II. CAPTURING AND PREPROCESSING

### A. Capturing

Machine is designed to place each specimen on a proper, white colored, surface (specimens are of dark color). Camera position will be zenithal so that we get a whole image of the flat fish. We will use LED panels for lighting (backlight LED panels with a diffuser). We will use a matrix (field) camera like JAI CM-200 1/1.8" CCD GigE[1].

Nowadays, we have developed computer vision algorithms using color images that were captured manually at a fish processing plant (figure 1).



Fig. 1. Test image (color has been enhanced through histogram stretching).

### B. Preprocessing

Image preprocessing will consist of segmenting the specimen. We start by binarizing image using the well-known Otsu threshold [2]. This method computes optimum threshold for a bimodal image (histogram has only two dominant peaks) using conditioned variances.

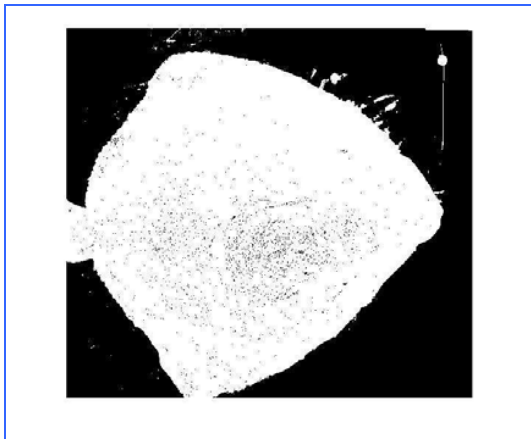


Fig. 2. Binarized image, after applying Otsu method.

After binarization, image is cleaned removing small objects that are not part of the specimen. This is achieved using mathematical morphology [3]. Now, we describe process, step by step:

- Opening with a circle of radius equal to 10 (this is computed for 2000 lines images, for other sizes the structuring element size can be rescaled). Purpose: erasing small objects.
- Closing with a circle of radius 5. Purpose: filling holes inside the fish mask.
- Labeling and extracting the biggest connected object (maximum area → dominant object).

Final result is like this:

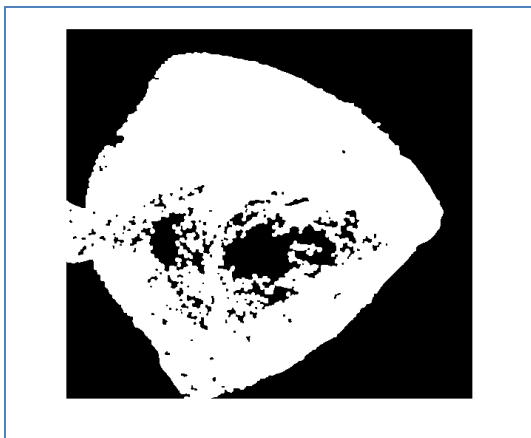


Fig. 3. Preprocessing result.

Remaining holes are not important (besides, they would be easily removed with the function “imfill”: image filling) but the contour and the fact that there is no noise.

Using the former image, it is easy to obtain the region of interest (usually called ROI in computer vision) that, in this case, it is defined by the front part including head. The nose is detected as the rightmost active (white) point.

Besides, figure 3 (specimen mask) allows computing easily the specimen contour. We define contour in this case as:

<<The curve that is made with the last (rightmost) active (white) point on each mask line.>>

Contour obtained in this manner has some discontinuous points (last point in a given line is not contiguous to that of the next one). This is correcting connecting those points with straight (horizontal) lines.

### III. CUTTING CURVE COMPUTATION

Once we have the contour, we use Hough transform [4] (geometric forms detection, straight lines in this case) to find the two dominant straight lines. Id EST: the one descending from the upper part until the nose and the one that ascends from bottom to the same point (nose). With the equations of these two lines, we will find the critical points in the contour. More precisely, we compute distance between each point in the contour and the nearest straight line (from the two referred ones).

That distance is processed as a signal, so that we can detect (figure 4):

- Beginning of the head (upper arrow). It is defined by the first local maximum of distance function. In fact, it is the local maximum that is nearest to the beginning or to the ending of such signal. Detecting this point, we know in what position we have specimen’s eyes. Laying in this manner (eyes up and nose looking right) it is more frequent that the eye that is closest to the contour is in the lower position, but some specimens have it reversed. For maxima detection, we have used “neighborhood peak detection” that is an algorithm designed by ourselves, published in [5].
- Once we have detected the first point, we analyze the other half curve (beyond the central point). We will search for the first local minimum (using the same method in [5]). This will allow us to detect head ending which coincides with an eye (lower arrow).

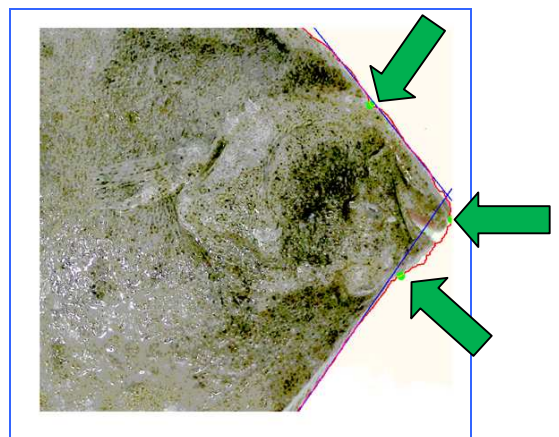


Fig. 4. Critical points detection.

The third critical point is the nose that was already known from the preprocessing stage. In this last image, we have applied a gamma correction that allows distinguishing better the color texture in the fish body, while “burning” to the extreme the white background. Nevertheless, detecting head contour using textures seems impossible.

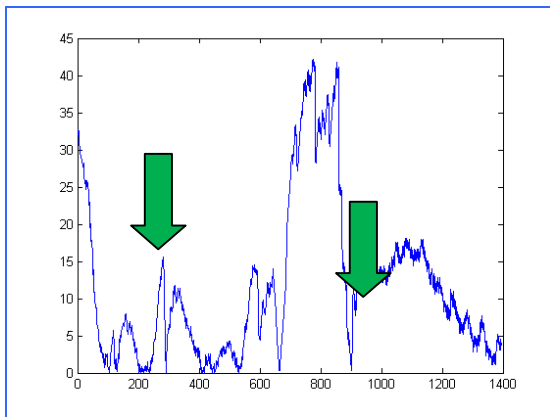


Fig. 5. Error signal between the contour and the two straight lines (figure 4). Horizontal axis is simply labeled with the sample count; the central value corresponds approximately to the nose. Vertical axis measures distance from contour points to the straight lines, units are simply pixels.

To compute the cutting curve, we decided to calculate a parabola that should pass by the two head endings (upper and lower critical points, computed just now). As it is a second order equation, we need three points to solve for the polynomial coefficients. Third point was heuristically determined from the three points in figure 4.

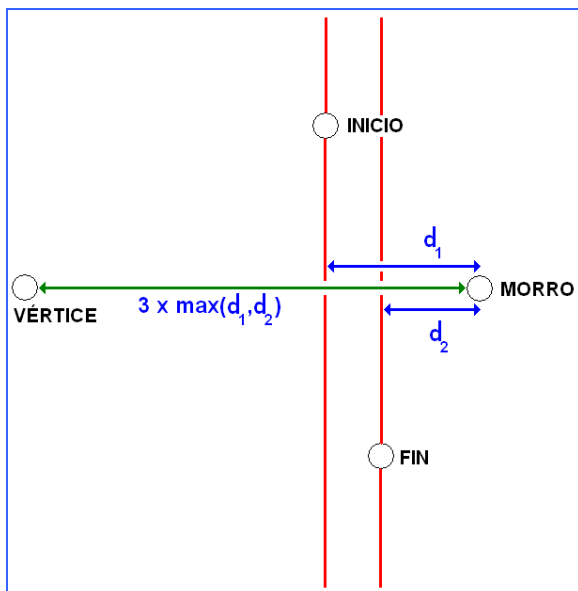


Fig. 6. Third point (vertex) computation.

The third point will be “approximately” the vertex of the parabola. It will be at the same height than nose and at a horizontal distance of it equal to three times  $d$ . Let  $d$  be the maximum between the two horizontal distances with head beginning and ending points. Calculation is illustrated in figure 6 and result is shown in figure 7.



Fig. 7. Cutting parabola.

This method has been developed and tested in Matlab [6] environment and has already been translated into C++ using the open source library OpenCv [7].

#### IV. TESTS AND RESULTS

Matlab application has been successfully tested on 27 images, 24 of them were obtained by us in the processing plant, the remaining 3 images (with lower quality) were provided by the cutting machine manufacturer (the client).

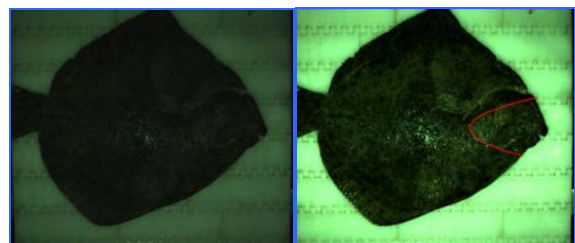


Fig. 8. Left: low quality image (with green lighting), Right: processed image, a contrast enhancement was performed before using the regular (previously described) method.

## V. CONCLUSIONS AND FUTURE LINES

We have found an easy yet effective method for computing automatically the cutting curve for turbot fish head. This is the solution for a previously unsolved problem.

Main future line is the integration with a final machine that, at the time of writing, still does not exist. Cutting curve must be sent from the industrial computer that runs the computer vision process to a robotic arm [8] that will cut the head. Communication will be possible using a XML file with a set of 2D points defined via spatial coordinates (x,y) measured in millimeters. In the final machine, specimen's nose will always be in the same point as mechanical devices will place it touching a metal piece with angle (or cradle) shape. This point will be the coordinate origin.

## ACKNOWLEDGEMENT

Authors want to acknowledge the "Centro para el Desarrollo Tecnológico Industrial" (CDTI, [www.cdti.es](http://www.cdti.es)) for their funding, project: CO-0170-11.

## REFERENCES

- [1] <http://www.jai.com>
- [2] N. Otsu, A Threshold Selection Method for Gray-Level Histograms, IEEE Transactions on System, Man and Cybernetics, vol. 9, N.1 (1979).
- [3] R.C. González, R.E. Woods, "Digital image processing using MATLAB", Steven L. Eddins, Upper Saddle River: Gatesmark Publishing, cop. 2009.
- [4] R.O. Duda, P. E. Hart, Use of the Hough Transformation to Detect Lines and Curves in Pictures, Comm. ACM, Vol. 15, pp. 11-15 (January, 1972).
- [5] F. Martín-Rodríguez, New Tools for Gray Level Histogram Analysis, Applications in Segmentation, Lecture Notes in Computer Science, Vol. 7950, pp. 326-335 (June, 2013).
- [6] <http://es.mathworks.com/products/matlab/>
- [7] <http://opencv.org/>
- [8] <http://www.kuka.com>