# Instruction and documentation

## For

**Code for recognition of the liquid surface and the liquid level in image of liquid in transparent vessel. For surfaces between liquid and air and between phase separating liquids**

**Base on paper**

**Computer vision-based recognition of liquid surfaces and phase boundaries in transparent vessels, with emphasis on chemistry applications**

This document contains description of the source code for recognition of liquid surface and levels in an image of liquid in transparent containers. The method described in the paper “Computer vision-based recognition of liquid surfaces and phase boundaries in transparent vessels, with emphasis on chemistry applications” available at arxiv: <http://arxiv.org/abs/1404.7174>.

C:\Users\mithycow\Desktop\GLassWare LAB PICTURES\Liquid Phases\Real results\Pictures used for liquid phase recogntion  experiments - Copy\two phase\IMG_3262_GrayScale.tifC:\Users\mithycow\Desktop\GLassWare LAB PICTURES\Liquid Phases\Real results\Pictures used for liquid phase recogntion  experiments - Copy\two phase\IMG_3262_Iborder.tifC:\Users\mithycow\Desktop\GLassWare LAB PICTURES\Liquid Phases\Real results\Pictures used for liquid phase recogntion  experiments - Copy\two phase\IMG_3262_Marked_Liquid_Phases_all.tif

**Figure S1. Input and output for the *Liquid\_Surface\_Line\_Recognition* function. Left: Liquid container image. Center: container boundaries binary image. Right: the results of the phase boundaries recognition marked on the image.**

The source code itself contain embedded explanation and documentation. This document could be used in addition. The main function responsible for the liquid surface recognition is Liquid\_Surface\_Line\_Recognition (Is,Iborder,outname….). This function recognizes liquid surfaces and levels in an image of liquid container Is (Figure S1 left). Where the boundaries/contour line of the vessel in the image are given in Iborder (Figure S1 Center). See code for finding vessel boundaries to see how Iborder could be created:

<http://www.mathworks.com/matlabcentral/fileexchange/46887-find-boundary-of-symmetric-object-in-image>

or

<http://www.mathworks.com/matlabcentral/fileexchange/46907-find-object-boundaries-in-image-using-template--variable-image-to-template-size-ratio->

The first parameter Is is a color image of the vessel containing liquid (Figure S1 Left). The second parameter Iborder is binary image that contain the contour of the vessel in the image Is (Figure S1 Center). The in pixels Iborder corresponding to the vessel contour are marked as one (white) the rest of the pixels marked as zero (black, Figure S1 Center). If Is and Iborder are not in the same size Is will be resized to the size of Iborder. The rest of the parameters of this function are all optional (could be left blank) and discussed in last section of this document (Also discussed the paper and in the function explanation embedded in the source code). As output Liquid\_Surface\_Line\_Recognition function write the image Is with the all the boundaries of the liquid surfaces recognized marked in black (Figure S1 Right), in directory and file name given by string in outname (For example “C:\image”). All recognition methods discussed in the paper used Liquid\_Surface\_Line\_Recognition function as the main function (The code in this function explained in section 2 of the paper). The difference between Entries/methods for recognition of liquid surface is the method for rating of the curve (Section 2.2, 3 of the paper). The rating of curve in the Liquid\_Surface\_Line\_Recognition function is done by functions MatchEllipse… in line 124(probably) of the Liquid\_Surface\_Line\_Recognition function. There various of such functions which use different methods to rate the curve, and can receive different inputs. The Connection between the function used to rate curve and the Entries in table 1-2 of the paper is given in table S1-S4 below (it is also given in the code documentation). Description of all main functions appear in the last section of this paper.

See the directory “EXAMPLE IMAGES” located in the source code directory for examples input and output images.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entry | Indicator Description | Local score equation  (Score for single curve point) | Image type  (Pixel values in local score equation) | Evaluation method for curve score from local scores | Method (section 3.1) | Threshold 0. | Name of function used for curve rating (In the source code supplied) | Embedded Consistency check in the function |
| 1 | Relative intensity change normal to curve (percentile) |  | Grayscale | Percentile (65%) | 1 | .4 | *MatchEllipse11* | No |
| 2 | Relative intensity change normal to curve (average) |  | Grayscale | Average | 1 | .4 | *MatchEllipse1* | Yes |
| 3 | Intensity change normal to curve |  | Grayscale | Percentile (65%) | 1 | .3 | *MathcEllipse12* | No |
| 4 | Global relative intensity change normal to curve1353 |  | Grayscale | Average | 1 | .4 | *MatchEllipse2* | Yes |
| 5 | Absolute intensity change normal to curve |  | Grayscale | Average | 1 | .4 | *MatchEllipse3* | Yes |
| 6 | Absolute relative intensity change normal to curve |  | Grayscale | Average | 1 | .4 | *MatchEllipse4* | Yes |
| 7 | Average intensity of pixel on curve |  | Grayscale | Average | 2 | .75 | *MatchEllipse7* | No |
| 8 | Relative difference between average intensity above and on curve |  | Grayscale | As it is | 3 | .4 | *MatchEllipse5b* | No |
| 9 | Normalized difference between average intensity above and on curve |  | Grayscale | As it is | 3 | .4 | *MatchEllipse5* | No |
| 10 | Difference between average intensity above and on curve |  | Grayscale | As it is | 3 | .4 | *MatchEllipse6* | No |
| 11 | Difference between average intensity inside and around surface curve453436 | (Ellipse interior intensity) - (Ellipse outer curve intensity) | Grayscale | As it is | - | .4 | *MatchEllipse8* | No |
| 12 | Relative edge density change normal to curve |  | Edge | Average | 1 | .5 | *MatchEllipse1* | Yes |
| 13 | Average edge density on curve |  | Edge | Average | 2 | .6 | *MatchEllipse7* | No |
| 14 | Edge density and scalar product of gradient direction and curve normal |  | Edge | Average | 2 | .45 | *MatchEllipse9d* | No |
| 15 | Difference between average edge density above and on curve |  | Edge | As it is | 3 | .4 | *MatchEllipse6* | No |
| 16 | Score of the curve in generalized Hough transform (12 angle bins) | Hough transform curve score | Edge | Average | - | .3 | *MatchEllipse15* | No |
| 17 | Change in gradient size normal to curve |  | Gradient Size | Average | 1 | .4 | *MatchEllipse13* | No |
| 18 | Average gradient size on curve |  | Gradient Size | Average | 2 | .5 | *MatchEllipse7* | No |
| 19 | Scalar product of gradient and curve normal |  | Gradient Size | Average | 2 | .5 | *MatchEllipse10d* | No |
| 20 | Difference between average gradient size above and on curve |  | Gradient Size | As it is | 3 | .4 | *MatchEllipse6* | No |

**Table S1: Relation between Entries in the paper (Tables 1-2) and functions used for curve rating in the source code. The rating of the curve is done by specific *MatchEllipse* function depends on the method use in the specific Entry(See sections 3-4 in the paper). Function name is the name of the function used for rating the curve in the code (line~124 of the *Liquid\_Surface\_Line\_Recognition* function). Entry is the entry used in tables 1-2 of the paper. Embedded consistency check in the function, tell if the consistency check (Section 4.4.1 paper) is performed within the function or as separated function (ConsistencyFilter). If the consistency check embedded in the function it is performed by default (but could be disabled by changing the function code). The rest of the parameter’s explained caption of table 2 in the Paper. All Entries in this table were run without consistency check (internal or external).**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entry | Indicator Description | Local score equation  (Score for single curve point) | Image Type  (Pixel values in local score equation) | Evaluation method of curve score from local scores | Method (section 3.1) | Threshold 0. | Name of function used for curve rating (In the source code supplied) | Embedded Consistency check in the function |
| 21 | Intensity change normal to curve |  | Grayscale | Average | 1 | .3 | MatchEllipse13 | Yes |
| 22 | Relative intensity change normal to curve |  | Grayscale | Average | 1 | .4 | MatchEllipse1 | Yes |
| 23 | Global relative intensity change normal to curve13,53 |  | Grayscale | Average | 1 | .4 | MatchEllipse2 | Yes |
| 24 | Absolute intensity change normal to curve |  | Grayscale | Average | 1 | .4 | MatchEllipse3 | Yes |
| 25 | Absolute relative intensity change normal to curve |  | Grayscale | Average | 1 | .4 | MatchEllipse4 | Yes |
| 26 | Relative intensity change normal to curve in 1% range (height of curve surroundings is 1% of vessel height) |  | Grayscale | Average | 1 | .4 | MatchEllipse1 | Yes |
| 27 | Relative intensity change normal to curve in 2% range (height of point surroundings is 2% of vessel height) |  | Grayscale | Average | 1 | .4 | MatchEllipse1 | Yes |
| 28 | Average relative intensity change normal to curve in the Red, Green, and Blue channels of the RGB color image. |  | Color (R, G, B channels) | Average | 1 | .4 | MatchEllipse1 | No (consistency check is performed separately on the grey scale image |
| 29 | Edge density change normal to curve |  | Edge | Average | 1 | .4 | MatchEllipse5b | No (disabled) |
| 30 | Average edge density on curve |  | Edge | Average | 2 | .45 | MatchEllipse7 | No |
| 31 | Edge density and scalar product  gradient direction and curve normal |  | Edge | Average | 2 | .4 | MatchEllipse9d | No |
| 32 | Difference between average edge density above and on curve |  | Edge | As it is | 3 | .4 | MatchEllipse6 | No |
| 33 | Scalar product gradient and curve normal |  | Gradient size | Average | 2 | .4 | MatchEllipse10d | No |
| 34 | Difference between average gradient size above and on curve |  | Gradient Size | As it is | 3 | .32 | MatchEllipse6 | No |
| 35 | Relative gradient size change  normal to curve |  | Gradient Size | Average | 1 | .6 | MatchEllipse1 | Yes |

**Table S2: Relation between Entries in the paper (Tables 1-2) and functions used for curve rating in the source code. The rating of the curve is done by specific *MatchEllipse* function depends on the method use in the specific Entry(See sections 3-4 in the paper). Function name is the name of the function used for rating the curve in the code (Line~ 124 of the *Liquid\_Surface\_Line\_Recognition* function). Entry is the entry used in tables 1-2 of the paper. Embedded consistency check in the function, tell if the consistency check (Section 4.4.1 paper) is performed within the function or as separated function (ConsistencyFilter). If the consistency check embedded in the function it is performed by default (but could be disabled by changing the function code). The rest of the parameter’s explained caption of table 2 in the Paper. All Entries in this table were run with consistency check (internal or external).**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entry | Indicator Description | Embedded Consistency check in the function | Implementation of curve rating in *Liquid\_Surface\_Line\_Recognition* function | Consistency check applied? | External consistency check applied |
| 1 | Relative intensity change normal to curve (percentile) | No | MatchEllipse11(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 2 | Relative intensity change normal to curve (average) | Yes | MatchEllipse1(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 3 | Intensity change normal to curve | No | MatchEllipse12(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 4 | Global relative intensity change normal to curve | Yes | MatchEllipse2(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 5 | Absolute intensity change normal to curve | Yes | MatchEllipse3(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 6 | Absolute relative intensity change normal to curve | Yes | MatchEllipse4(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 7 | Average intensity of pixel on curve | No | MatchEllipse7(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 8 | Relative difference between average intensity above and on curve | No | MatchEllipse5b(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 9 | Normalized difference between average intensity above and on curve | No | MatchEllipse5(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 10 | Difference between average intensity above and on curve | No | MatchEllipse6(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 11 | Difference between average intensity inside and around surface curve453436 | No | MatchEllipse8(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 12 | Relative edge density change normal to curve | Yes | MatchEllipse1(Icanny,Iellipse,Resl,y(f),Iinterior) | No | No |
| 13 | Average edge density on curve | No | MatchEllipse7(Icanny,Iellipse,Resl,y(f),Iinterior) | No | No |
| 14 | Edge density and scalar product of gradient direction and curve normal | No | MatchEllipse9d(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 15 | Difference between average edge density above and on curve | No | MatchEllipse6(Icanny,Iellipse,Resl,y(f),Iinterior) | No | No |
| 16 | Score of the curve in generalized Hough transform (12 angle bins) | No | MatchEllipse15(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 17 | Change in gradient size normal to curve | No | MatchEllipse13(Igradient,Iellipse,Resl,y(f),Iinterior) | No | No |
| 18 | Average gradient size on curve | No | MatchEllipse7(Igradient,Iellipse,Resl,y(f),Iinterior) | No | No |
| 19 | Scalar product of gradient and curve normal | No | MatchEllipse10d(Igr,Iellipse,Resl,y(f),Iinterior) | No | No |
| 20 | Difference between average gradient size above and on curve | No | MatchEllipse6(Igradient,Iellipse,Resl,y(f),Iinterior) | No | No |

**Table S3: Relation between Entries in paper (Table 1-2) and functions used for curve rating the source code. The rating on of the curve is done by specific *MatchEllipse* function depending on the method used in the Entry (See section 3-4 paper). “Implementation of curve rating *in Liquid\_Surface\_Line\_Recognition function* Column give the function (*MatchEllipse*) used for calculating the curve score and its input parameters, used to rating the curve in the code (Line~ 124 of the *Liquid\_Surface\_Line\_Recognition* function). The input parameters for the *MatchEllipse* functions are: *Igr* is the greyscale image of the liquid vessel. *Igradient* is the sobel gradient map of *Igr*. Icanny is the canny edge image of *Igr*. *Iellipse* is binary image of the curve to be rated, in this case some kind of horizontal ellipse (*Iellipse* is Given as binary image of the ellipse where the pixels of the curve are marked 1. The size of *Iellipse* is the same as the size of *Igr/Icanny/Igradient* image). *Resl* is the resolution of the scan, the number of pixels above and below the curve used in evaluating the curve score. Value of 1 for *Resl* give best results for most cases (Resolution discussed in section 4.5 of the paper) . *y(f)* is the Y axis vertical coordinates of the line examine in the image (*Igr*) coordinates, basically the Y values of the the ellipse center in *Iellipse*. *Iinterior* give the area of the vessel in the image (*Igr*). *Iinterior* is a binary image in the size of *Igr/Icanny* with pixels corresponding to the vessel interior in the marked as one and the rest zero. “Embedded consistency check in the function” column, tell if the consistency check (Section 4.4.1 paper) is performed within the function or as separated function (*ConsistencyFilter*). If the consistency check embedded in the function it is applied by default (but could be remove by changing the code). “External consistency check applied” column, say if consistency check is applied in the scan using external function to *matchellipse* (*ConsistencyFilter*). Entry column is the entry used in tables 1-2 of the paper. The rest of the parameter’s explained caption of table 2 in the Paper. All Entries in this table were run without consistency check.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entry | Indicator Description | Embedded Consistency check in the function | Implementation of curve rating in Liquid\_Surface\_Line\_Recognition function | External Consistency check applied |
| 21 | Intensity change normal to curve | Yes | MatchEllipse13(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 22 | Relative intensity change normal to curve | Yes | MatchEllipse1(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 23 | Global relative intensity change normal to curve13,53 | Yes | MatchEllipse2(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 24 | Absolute intensity change normal to curve | Yes | MatchEllipse3(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 25 | Absolute relative intensity change normal to curve | Yes | MatchEllipse4(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 26 | Relative intensity change normal to curve in 1% range (height of curve surroundings is 1% of vessel height) | Yes | MatchEllipse1(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 27 | Relative intensity change normal to curve in 2% range (height of point surroundings is 2% of vessel height) | Yes | MatchEllipse1(Igr,Iellipse,Resl,y(f),Iinterior) | No |
| 28 | Average relative intensity change normal to curve in the Red, Green, and Blue channels of the RGB color image. | No (consistency check is performed separately on the grey scale image | MatchEllipse1(Rchannel,Iellipse,Resl,y(f),Iinterior)+MatchEllipse1(Bchannel,Iellipse,Resl,y(f),Iinterior)+MatchEllipse1(Gchannel,Iellipse,Resl,y(f),Iinterior) | Yes |
| 29 | Edge density change normal to curve | No (disabled) | MatchEllipse13(Icanny,Iellipse,Resl,y(f),Iinterior) | Yes |
| 30 | Average edge density on curve | No | MatchEllipse7(Icanny,Iellipse,Resl,y(f),Iinterior) | Yes |
| 31 | Edge density and scalar product  gradient direction and curve normal | No | MatchEllipse9d(Igr,Iellipse,Resl,y(f),Iinterior) | Yes |
| 32 | Difference between average edge density above and on curve | No | MatchEllipse6(Icanny,Iellipse,Resl,y(f),Iinterior) | Yes |
| 33 | Scalar product gradient and curve normal | No | MatchEllipse10d(Igr,Iellipse,Resl,y(f),Iinterior)) | Yes |
| 34 | Difference between average gradient size above and on curve | No | MatchEllipse6(Igradient,Iellipse,Resl,y(f),Iinterior) | Yes |
| 35 | Relative gradient size change  normal to curve | Yes | MatchEllipse1(Igradient,Iellipse,Resl,y(f),Iinterior) | No |

**Table S4: Relation between Entries in paper (Table 1-2) and functions used for curve rating the source code. The rating on of the curve is done by specific *MatchEllipse* function depending on the method used in the Entry (See section 3-4 paper). “Implementation of curve rating *in Liquid\_Surface\_Line\_Recognition function* Column give the function (*MatchEllipse*) used for calculating the curve score and its input parameters, used to rating the curve in the code (Line~ 124 of the *Liquid\_Surface\_Line\_Recognition* function). The input parameters for the *MatchEllipse* functions are: *Igr* is the greyscale image of the liquid vessel. *Igradient* is the sobel gradient map of *Igr*. Icanny is the canny edge image of *Igr*. *Iellipse* is binary image of the curve to be rated, in this case some kind of horizontal ellipse (*Iellipse* is Given as binary image of the ellipse where the pixels of the curve are marked 1. The size of *Iellipse* is the same as the size of *Igr/Icanny/Igradient* image). *Resl* is the resolution of the scan, the number of pixels above and below the curve used in evaluating the curve score. Value of 1 for *Resl* give best results for most cases (Resolution discussed in section 4.5 of the paper) . *y(f)* is the Y axis vertical coordinates of the line examine in the image (*Igr*) coordinates, basically the Y values of the the ellipse center in *Iellipse*. *Iinterior* give the area of the vessel in the image (*Igr*). *Iinterior* is a binary image in the size of *Igr/Icanny* with pixels corresponding to the vessel interior in the marked as one and the rest zero. “Embedded consistency check in the function” column, tell if the consistency check (Section 4.4.1 paper) is performed within the function or as separated function (*ConsistencyFilter*). If the consistency check embedded in the function it is applied by default (but could be remove by changing the code). “External consistency check applied” column, say if consistency check is applied in the scan using external function to *matchellipse* (*ConsistencyFilter*). Entry column is the entry used in tables 1-2 of the paper. The rest of the parameter’s explained caption of table 2 in the Paper. All Entries in this table were run with consistency check.**

# 2. Description of some other functions used in the code

The following section contains brief explanation of main functions in the code (these functions also contain documentation embedded in the code itself).

***Liquid\_Surface\_Line\_Recognition(Is,Iborder,outname,Mode,MinRes,MinWidth,MinScore, Hightd\_To\_Width\_Ellipse\_Ratio,MinFractScore)***

The main liquid surfaces recognition function.

Given image *Is* of liquid in vessel (in color) and binary edge image *Iborder* in which pixels on the boundary of the vessel region in the image *Is* have value of 1.

The function recognize the borders and surfaces of each liquid surface inside the vessel, this in include the top and floor of the vessel (which are always recognise as surfaces).

As output the function write the recognised liquid-surface/phase-boundaries marked in black on the original image *Is*. It save this image as tif image in directory and name given by text sting *outname*

Other than *Is, Iborder, outname* all parameters are optional and could be left blank.

This function algorithm is explained in section 2 of the paper.

**Parameters**

(See section 2.1 of the paper for this parameters)

***Is*** is the original image of the liquid container in color.

***Iborder*** is the edge image (binary image) of the liquid container contour in image *Is*. Pixels in ***Iborder*** corresponding to the boundaries of the of the vessel in image *Is* have value of 1, and the rest of the pixels have value of zero.

(Iborder can be created as the \_BORDERS.tif output of the code in:

<http://www.mathworks.com/matlabcentral/fileexchange/46887-find-boundary-of-symmetric-object-in-image>

or

<http://www.mathworks.com/matlabcentral/fileexchange/46907-find-object-boundaries-in-image-using-template--variable-image-to-template-size-ratio->

***MinFractScore*** is the minimal threshold score for curve to be accepted as surface compared to the best score achieve by all curves in the scan. The threshold score for accepting curve as liquid surface will be *[Threshold Score]=* *MinFractScore•[Best score for all curve scanned]*. See section 2.3 in the paper for more details.

***MinScore*** is the minimum curve score needed for curve in to be accepted as liquid surface (this can used as alternative or in addition to *MinFractScore* ) it could be set to zero if *MinFractScore* used. *MinScore* is basically the threshold score for accepting curve as liquid surface. All curves with score higher than *MinScore* will be accepted. See section 2.3 in the paper for more details.

***MinWidth*** is the minimal width of image regions in the vessel that will be scanned (as fraction of the maximal width of the vessel) in the image. Areas of the vessel narrower than *MinWidth•[Maximal vessel width]* pixels will not be scanned (see section 2.1 of paper).

***Width\_To\_Hight\_Ellipse\_Ratio*** give the maximal height of the ellipse scanned compared to the width of that line/ellipse. The larger this parameter the lower the maximum ellipse height will be in the scaned. Values between 0.2 to 0.4 are recommended, values of one is the maximal which mean all possible ellipse from line to circle will be scanned for *Width\_To\_Hight\_Ellipse\_Ratio=1*.

***MinRes*** is the minimal resolution in which the scanning will be done. Hence the number of lines (in pixels) above and below the curve line that will be used for evaluating the curve score. Value of 1 for *MinRes* give best results for most cases.

***outname*** is text string in which the directory and file name for the output image files are written. For example “C:\output\file1” will lead to output image C:\output\file1.tif.

**Function description.**

**(See section 2 of paper).**

General what this function do is scan line by line in the area of the image belong to the vessel (Areas in *Is* inside the contour of *Iborder*).

For each line it generated various of elliptic curves that correspond to possible shape of liquid surface centred in this line.

The curves are compared to the image to find score for its correspondence to liquid surface in the image.

The liquid surface must look like either straight horizontal line or horizontal ellipse in order to be recognised. The curves with scores that pass some threshold given by *MinFractScore* or *MinScore* are accepted and marked on the image, which is saved as TIF image in location given by *outname*.

----------------------------------------------------------------------------------------------------------------------

***[Score] = MatchEllipseXXX(Ir,Iellipse,Resly,Ycnt,Ierea)***

Include group of functions, where *XXX* is some number correspond to the specific curve rating method.

These functions rate of the correspondence between the curve in *Iellips*e and liquid-surface/phase-boundary in image *Ir* a and return *score* belong to this correspondence (The algorithm described in section 2.2 and 3 of the paper).

**Parameters:**

***Ir*** is the image of the liquid surface which could be grayscale image, Edge image or gradient image.

***Iellipse***  is binary image of the curve to be rated, in this case some kind of horizontal ellipse or horizontal line (*Iellipse* is Given as binary image of the ellipse where the pixels of the curve are marked 1, and all the rest of the pixels have value of 0. The size of *Iellipse* is the same as the size of *Ir*).

***Resly*** is the resolution of the scan, the number of pixels above and below the curve used in evaluating the curve score (not all *MatchEllipse* functions can use this parameter). Value of 1 for *Resl* give best results for most cases (Resolution discussed in section 4.5 of the paper) .

***Ycnt***is the Y axis vertical coordinates of the center line of the curve (in *Iellipse*). Basically the Y values of the ellipse center in *Iellipse*.

***Iinterior*** give the region of vessel in the image (*Igr*).  *Iinterior* is a binary image in the size of *Ir* with all pixels corresponding to the vessel interior in the original image marked as one and the rest of the pixels marked as zero. For working mode see section 3,4 of the paper.

----------------------------------------------------------------------------------------------------------------------

***[Score] =ConsistencyFilter(Ir,Iellipse,Resly,Ycnt,Ierea,Fract, Thresh)***

This check the consistency of the relative intensity change normal to the curve ***Iellipse*** in image ***Ir*** (mostly grayscale image). The function return 1 if more than ***Fract*** percent of the curve point show relative intensity change that pass ***Thresh*** value to the same direction (with the same sign). The rest of the parameters are the same as in *MatchEllipse*.

See section 4.4.1 of the paper.

----------------------------------------------------------------------------------------------------------------------

***[Iel,sumofpixels] = ELLIPSE( x1,x2,y1,y2, posdil,negdil,sizeim, PartsToDraw )***

Draw specific elliptic curve in an binary image (*Iel*). Create binary image with the ellipse drawn on it.

***x1,x2,y1,y2*** are the leftmost rightmost top and bottom coordinates of the ellipse respectively.

***Sizeim*** is the size of the binary image(*Iel*) in which the ellipse should be drawn (should be the same as the size of liquid container image).

***posdil*** and ***negdil parameters*** are not being used and should be set to zero. This parameters controls the positive and negative dilation of the ellipse curve generated.

***PartsToDraw*** again parameter that isn’t being used and could be left blank. This parameter determines if to draw all the ellipse or only the top/bottom half. Draw all ellipse by default.

***Iel***The output binary image with the ellipse drawn. Pixel belong to ellipse line have value of 1. The rest have value of zero. The size of *Iel* image is *Sizeim****.***

***Sumofpixels*** is the sum of all pixel in *Iel* which is basically the number of point on the ellipse drawn. This output parameter is not used.

----------------------------------------------------------------------------------------------------------------------

***[y,x1,x2,np]=find\_binary\_contour\_leftright\_edges(BW)***

This function take the binary image ***BW*** that contain the contour of the vessel (Figure S1 center), and find the left and right edges (in image coordinates) of every line inside the contour given in *BW*.

***x1[n]*** array that contain the left most *x* coordinate of line *n*, in the vessel region of the image.

image).

***x2[n]*** array that contain the right most x coordinate of line *n*, in the vessel region of the image.

***y[n]*** array that contain the y coordinate of line *n*, in the vessel region of the image.

***np*** the number of lines in the vessel region of the image.

----------------------------------------------------------------------------------------------------------------------

***Directory\_Liquid\_Phases\_Recognition.m***

This script basically preform liquid surface recognition on every jpg image file in a given directory (*SystemDir*, line 24)

The directory must contain color images of the liquid containing vessel in jpg format (this format be change by changing “.jpg” in line 32)

For every color jpg image the directory most contain a binary edge images of the vessel contour in the liquid vessel.

The name of the edge file must be the same as the name of the liquid image for which this boundaries belong + \_BORDERS.tif extension . For example “v1.jpg” and “v1\_BORDERS.tif”. Images that do not have contour files with \_BORDERS extension will be ignored.

Basically the script scan for all files with color image of the system directory *SystemDir*

and find files with equivalent name which end by \_Border.tif.

It then transfer this to files to the function *Liquid\_Surface\_Line\_Recognition* which performs the recognition.

**Example images**

Several example input and output files are given in the directory “EXAMPLE IMAGES” in the located source code directory.