HOKKAIDO UNIVERSITY

Title	An image processing algorithm for detecting in-line potato tubers without singulation
Author(s)	Al-Mallahi, A.; Kataoka, T.; Okamoto, H.; Shibata, Y.
Citation	Computers and Electronics in Agriculture, 70(1), 239-244 https://doi.org/10.1016/j.compag.2009.11.001
Issue Date	2010-01
Doc URL	http://hdl.handle.net/2115/42631
Type	article (author version)
File Information	CEA70-1_239-244.pdf

Instructions for use

Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

An Image Processing Algorithm for Detecting in-Line Potato Tubers without Singulation

A. Al-Mallahi[1], T. Kataoka[2], H. Okamoto[2], Y. Shibata[2]

[1]Graduate School of Agriculture, Hokkaido University, Kita-9, Nishi-9, Kita-ku, Sapporo Hokkaido 060-8589, Japan, Email: ahmad@bpe.agr.hokudai.ac.jp, Tel: +81117063831, Fax: +817062555

[2] Research Faculty of Agriculture, Hokkaido University, Kita-9, Nishi-9, Kita-ku, Sapporo Hokkaido 060-8589, Japan

Abstract

An algorithm which helps detecting in-line potato tubers without singulation when they are mixed with clods was developed. It is particularly useful for sorting and grading purposes whether at the packhouses or the potato harvester. The algorithm scans images which include clusters of tubers and clods from left to right until it encounters a cluster. Once encountered, a scan around the periphery of the cluster starts until returning to the original point where the scanning around the periphery started. At the end of the periphery scanning, the periphery pixels are erased and the new width of the cluster is compared with the previous width. Contouring the periphery continues until a sudden shrink in the width of the cluster occurs which indicates the detection of a contact point. The left-to-right scanning resumes from the contact point until it reaches the right end of the image repeating the periphery scanning every time a cluster is encountered.

The algorithm was tested by detecting 380 contact points between tubers and 112 contact points between tubers and clods. Results showed that approximately 99% of the contact points were detected successfully. It was found that the only reason of error was the lack of cusp between a

tuber and a clod when the clod did not have an oval shape. It could be concluded that the algorithm is able to detect any contact point between any oval shaped adjacent objects as long as they form a cusp between them, which is guaranteed between any adjacent tubers. Therefore, the algorithm can be applied at the packhouses to replace the long, bulky and expensive conveyors needed to create spaces between the in-line potato tubers so as to aid machine vision-based size grading. Moreover, the algorithm can be used at the potato harvester to help in discriminating between the touching tubers and clods so as to automate the removal of the clods during harvesting

Keywords: Image Processing, Potato Harvester, Machine Vision, Singulation, Clod

Nomenclature			
		y	The order of the pixel in the vertical axis of the image from up to down
x	The order of the pixel in the horizontal axis of the image starting from left to right	y'	The vertical location from which scanning the image starts from left to right

1. Introduction

In fruit and vegetable packhouses, agricultural products flow in big quantities on conveyor belts at which they are usually clustered or piled. For sorting and packing purposes, the products should be singularized either mechanically or manually before being graded based on size, quality or both. Singulation mechanisms consist of series of conveyor belts which increase in width and speed gradually until the objects get detached from each other before heading toward sorting mechanisms [1, 2]. Most of the Current sorting mechanisms operate based on machine vision systems where one image or more are taken for each individual object after being singularized and the result of processing the images will decide the size and the quality of the object [3, 4, 5].

Singulation mechanisms occupy large areas in any packhouse and manufacturing and maintaining this long and large series of belt conveyors could be costly and cumbersome. Therefore, there have been many attempts to get rid of or, at least, to shrink the size of the mechanisms to become more reasonable in cost and size [6, 7].

Potato is one of the products which sorting takes essential part in its harvesting and post-harvest operations. Potato harvesters carry tubers, mixed with clods, from ground to storage containers which means that the clods should be removed on the spot during harvesting. Also, damaged and

green tubers should be removed not to harm the sound tubers if they are stored together. Finally, the sound tubers are sorted to different sizes in the packhouses before being packed [8].

The lack of a compacted singulation mechanism is one of the reasons, which obstruct automating the removal of the clods and the damaged potatoes on the harvester. Marchant et al [7] developed a blob splitting algorithm which separated touching potatoes, arranged in lines, to upper and lower parts and investigated the boundaries of each part. The contact point between two potatoes was considered, using this algorithm, to be in the area where opposing cusps from the upper and lower parts were approaching if the distance between the cusps were greater than a threshold. Hofstee and Molema [9] used the blob splitting algorithm in their attempt to estimate potato volumes for an online yield measuring system. They found that the algorithm was successful in splitting the cluster of potatoes consisted of up to 4 tubers. However, the dirt covering the tubers was another problem which needed to be solved for a successful detection of the potatoes online. Gogenini et al [10] developed an online yield monitoring system for estimating sweet potatoes using a single camera inside a chamber. The system was successful in estimating the sweet potatoes in the experimental and actual levels but yet the existence of mud would obstruct clear vision of the potatoes and the system was recommended to be applied on dry soil. Finally, Al-Mallahi et al [11] developed a vision system to discriminate between potato tubers and clods for an online clod removal mechanism which was able to distinguish potato tubers despite the existence of clods. However, the existence of clusters of clods and tubers limited its performance and a singulation mechanism was concluded to be essential for optimum performance of the vision system.

In this research, a method for potato and clod detection without singulation was developed. The objective was to enable the detection of the contact points in clusters of only potatoes or in clusters of potatoes and clods mixed together.

2. Materials and Methods

2.1 Vision System

Video images of clusters of tubers and clods passing in lines were taken on the conveyor of a potato harvester. Each line consisted of two to four objects whether touching each other forming a cluster or with gaps. The video images were taken using monochrome ultra violet camera (Sony, XC-EU50) which has a Pentax (B2528-UV) attached lens. The video frames were sent via a TV capture device (Buffalo, PC-SMP2E) to a computer (IBM-X61, 2.1GHz) for processing. Figure 1 shows a sketch of the devices arrangement while Figure 2 (a) shows an example of how the objects appear in the video frames before processing takes place.

Al-Mallahi et al [12] used the same devices to develop a vision system for discriminating between tubers and clods in 3 steps by developing a software written in the C# language. The first step was to isolate the objects from the background, the second step was to segment each pixel in the image independently as a tuber, clod or background and the third step was to segment each object by counting the number of pixels in each cluster of pixels to determine whether the cluster was a tuber or a piece of clod based on a majority rule. The advantage of this system was its ability of detecting tubers even if they were covered, partially, by mud or soil. However, the disadvantage was its lack of ability to discriminate the beginning and the ending of each object if they were touching each other. Figure 2 (b) to (d) show the steps in discriminating between the tubers and the clods, where it can be noticed in Figure 2(d) that one of the clods was misclassified because it was touching the two tubers. Therefore, an additional step of

singularizing the inline objects, by detecting their contact points and drawing vertical black lines on them, was inserted between the pixel and object segmentations so that the object segmentation could be applicable on each object independently.

2.2 Singulation Algorithm

The developed singulation algorithm searches for the contact point between two adjacent objects by continuous contouring around the periphery of any cluster. This process continues until the horizontal diameter of the cluster shrinks to smaller than a specific threshold which is set to be equal to the smallest possible diameter of potato tubers.

First, the algorithm starts scanning from left to right until it finds a non-black pixel from a designated point in the *y*-axis (0, *y'*) . Once encountered, it will be considered a periphery pixel and a search for a neighbouring periphery pixel begins based on 8-neighbour connectivity and clockwise scanning [6]. A periphery pixel is defined to be a non-black pixel which has from 1 to 3 black neighbouring pixels based on 4-neighbour connectivity [13]. Figure 3 is a simplified sketch of two touching tubers while the enlarged part of the figure illustrates the detection of the first periphery pixel in a cluster and the search for its neighbour. The scanning process keeps searching for periphery pixels and adding them to an array of pixels, called periphery array, until returning back to the first periphery pixel. By this, the first contouring around the periphery of the cluster will be completed and the pixels at which the maximum and minimum *x*-axis values were found, as well as the last pixel in the periphery array will be saved.

Next the periphery pixels in the periphery array will be cleared, causing the cluster to shrink 2 pixels in diameter, and a new search for the periphery will take place from a neighbouring pixel, called resuming point, to the pixel which has the minimum *x*-axis value in the periphery. The resuming point itself is found using a progressive scanning around the right side pixels of the

pixel of the minimum *x*-value in the previous scanning until finding a non-isolated periphery pixel as described in Figure 4.

The scan around the contour continues in the same way and every time the value of the maximum *x*-axis value in the current contour is compared to the maximum *x*-axis value from the previous contour. If a sudden shrink in the diameter of the cluster is detected, the pixel with the maximum *x*-value in the latest periphery will be considered as the contact point as illustrated in Figure 5, where a contact point between two touching objects is found after 3 scans around the periphery. The periphery scanning continues in the same way until the size of the cluster becomes very small, i.e. its diameter is smaller than the smallest possible diameter of a tuber.

Next, a black vertical line is drawn at the contact points found within the cluster and the line scanning from left to right resumes from the contact point with the smallest value. This process guarantees that all objects regardless of their sizes will be scanned around their peripheries and that all clusters will be scanned even if there where gaps between them. Figure 6 is a flowchart which shows the steps required to draw the black lines at the contact points.

2.3 Handling Isolated Pixels

Scanning the periphery of a cluster of objects until returning to the starting pixel is guaranteed as long as the shapes of the objects are regular. However, the facts that the tubers are covered by, and mixed with, pieces of clods and that the images being analyzed are video images, increase the irregularity in the shapes of the clusters generating isolated pixels. An isolated pixel is defined as any pixel which has 3 or 4 black neighbours based on 4-neighbour connectivity.

One reason for generating the isolated pixels is one-pixel-thick horizontal lines appearing on the peripheries of the objects while moving, due to the vibration of the conveyor. In order to

eliminate the effect of vibration, the pixel-segmented image is smoothened before applying the singulation algorithm. Smoothening takes place by scanning the entire image and deleting any isolated pixel based on 4-neighbour connectivity.

Cavities in clods are the other reason for generating isolated pixel. They existed because of the methodology used to obtain the pixel-segmented images in Al-Mallahi et al [12] which misclassifies, occasionally, some pixels in the clods as black background pixels. Cavities can be avoided by changing the methodology used to obtain the pixel-segmented image which is out of the scope of this paper. Nevertheless, an amendment to the algorithm was added to reduce the effect of cavities in interrupting the periphery scanning. If the periphery scanning encounters a cavity, it will be considered as a contact point and a black line will be drawn to split the clod which allows the left-to-right scanning to resume from the cavity point.

2.4 Validation Methodology

The ability of the algorithm to singularize the objects, i.e. to find the contact point, was examined by applying it to images which included objects in lines. 3 different kinds of contact points were examined. The first was the contact point between tubers when the clusters were consisted of tubers only. The second was the contact point between tubers as well but when the clusters consisted of tubers and clods mixed together. The third was the contact point between tubers and clods in a cluster mixed of both. However, the clusters which included clods were examined once more when the cavities in the clods were eliminated manually so as to evaluate the effect of the irregular shapes of the clods in causing misclassification.

Finally, the number of the in-line clusters in the image and the number of in-line objects in each cluster were varied from 2 to 4 objects. Varying the number cluster was used to evaluate the ability of the algorithm to find a new cluster after scanning a previous cluster. Whereas, varying

the number of objects within the cluster was used to evaluate the ability of the algorithm to find more than one touching point within one cluster.

3. Results and Discussions

Table 1 shows the number of contact points being examined at the different kinds and the different numbers of objects in the cluster. The denominator indicates the total number of contact points examined, whilst the numerator indicates the number of contact points detected successfully. Moreover, the number in brackets shows the number of contact points detected successfully before the elimination of the cavities when the cluster consisted of tubers and clods at the same time.

Most of the examined clusters consisted of 4 objects (61.6% of the total number of clusters). It was thought that being able to detect the contact points in large groups indicates, by default, the ability of the algorithm of detecting clusters in smaller groups.

When the clusters consisted of tubers only, 100% of the contact points in the clusters were found. Moreover, the left-to-right scanning was able to skip the gaps when more than one cluster were found because, once resumed after finding the contact points in a certain cluster, the left-to-right scanning starts from the vertical point where the first contact point was found. Figure 7 shows how the algorithm could detect the contact points in the cluster of Figure 2.

The contact points between tubers, observed when the clusters were mixed with clods were detected before and after eliminating the cavities. In the former, approximately 93% of the contact points were found, while in the latter 100% of the contact points were successfully found. Failing to find the contact points when cavities existed, occurred only when the cavity itself was not in-line with the following object, i.e. when a clod was bigger in size than a right-side

neighbouring object and the cavity encountered was in the edge of the clod which caused the left-to-right scanning to skip the right-side objects.

The percentages of finding the contact points between tubers and clods were approximately 91% and 97% when cavities existed and when they were eliminated, respectively. The high percentages suggested that the scanning around the peripheries of the clods was successful most of the times despite the extensive irregularity of the clods. Splitting the clods when cavities were detected allowed the left-to-right scanning to continue even when isolated pixels were found.

In the clusters where contact points were not detected even when there were no cavities, failing to find the contact points was due to the non-oval shape of the clods. It could be said that the existence of a cusp, no matter how small it would be, is the only condition required to detect the contact points between the clods. Cusps may not be formed between tubers and clods if the clods are wet and attached, rather than just touching, to the surface of the tubers. On the other hand, due to the oval shape of the tubers, cusps will always be present which guarantees the success of finding their contact points each time they are searched for by the algorithm.

The wide range of sizes of the tubers did not affect the performance of the algorithm. The only assumption required was that the tuber should have a minimum width of approximately 3 cm. This was equivalent to 20 pixels in the images taken during the experiments.

Increasing the number of objects in the cluster did not affect the ability of the algorithm to detect the contact points, as it can be observed from the detection rate of finding the points between the tubers. Moreover, the contact points between any oval shaped objects of any size could be detected. Accordingly, this shows that the developed algorithm could be extended in its applications to detect other kinds of fruits and vegetables which require grading systems similar to the one used for the potatoes.

4. Conclusion

This paper aimed to develop an algorithm able to singularize potato tubers and clods arranged in lines. The developed algorithm is based on the principle of scanning images from left to right until encountering a cluster, around which a periphery scanning starts until a sudden shrinkage in its width occurs, indicating the existence of a contact point. A vertical black line is drawn at the contact point, from where a new search for another contact point resumes until reaching the right end of the image.

The results of testing the algorithm showed that 100% of the contact points between the tubers and 97% of the contact points between the tubers and the clods were found successfully. It was found that the contact point between any two oval objects could be found as long as a cusp is formed between them, because all oval clods were also detected despite their irregular shapes. The results also showed that the left-to-right scanning of the image was accomplished in each time it was tested. Having cavities of black pixels inside any cluster was found to be the major reason for reducing the success rate of detecting the contact points. An amendment was added to the algorithm to overcome this problem and increase the success rate. However, avoiding the generation of cavities by changing the methodology of isolating the objects from the background and obtaining cavity-free pixel-segmented images is recommended to obtain reliable performance of the algorithm.

The ability of the algorithm to detect the contact points among oval objects is promising to contribute in improving the detection abilities of machine vision systems developed for agricultural applications. It can be applied at the packhouses to replace the long, bulky and expensive conveyors needed to create spaces between the in-line potato tubers so as to aid machine vision-based size grading. Moreover, the algorithm can be used in the potato harvester

to help in discriminating between the touching tubers and clods so as to automate the removal of the clods during harvesting.

Acknowledgements

The authors gratefully acknowledge the Japan Society for the Promotion of Science (JSPS) for the Grants-in-Aid for Scientific Research (No. 18580251).

References

[1] E.G. Humphries, Vegetable Transport Operations with Horizontal Roller Conveyors, Applied Engineering in Agriculture. American Society of Agriculture Engineers, 17(1) (2001), 5-7.

[2] N. Kondo, K. Ninomiya, J. Kamata, Eggplant Grading Machine by Use of Rotary Trays, Proceedings of the 7-8 October 2004 Conference ASAE. 701P1004.

[3] J. Blasco, N. Alexios, E. Molto, Machine Vision System for Automatic Grading of Fruit. J. Biosystems Engineering, 85 (4), (2003), 415-423.

[4] S. Laykin, V. Alchanatis, E. Fallik, Y. Edan, Image Processing Algorithms for Tomato Classification, Transaction of ASAE, 45 (3), (2002), 851-858.

[5] M.A. Shahin , E.W. Tolner, R.W. McClendon, H.R. Arabnia, Apple Classification Based on Surface Bruises Using Image Processing and Neural Network, Transaction of the ASAE. 45 (5), (2002), 1619-1627.

[6] P.H. Heinemann, N.P. Pathare, C.T. Morrow, An Automated Inspection Station for Machine-Vision Grading of Potatoes. Machine Vision and Applications. 9, (1996), 14-19.

[7] J.A. Marchant, C.M. Onyango, M.J. Street, Computer Vision for Potato Inspection without Singulation, Computers and Electronics in Agriculture. 4, (1990), 234-244.

[8] D.C. McRae. A Review of Developments in Potato Handling and Grading, *J. of Agricultural Engineering Research*, 31, (1985), 115-138.

[9] J.W. Hofstee, G.J. Molema, Volume Estimation of Potatoes Partly Covered with Dirt Tare. ASAE Meeting Presentation, (2002), 031001.

[10] S. Gogineni, J.G. White, J.A. Thomasson, P.G. Thompson, J.R. Wooten, M. Shankle, Image-Based Sweet Potato Yield and Grading Monitor. 2002 ASAE Annual Meeting / CIGR XVth World Congress. Chicago, USA. 021169.

[11] A.A. Al-Mallahi, T. Kataoka, H. Okamoto, Discrimination between Potato Tubers and Clods by Detecting the Significant Wavebands, J. Biosystems Engineering. 100, (2008) 329-337.

[12] A.A. Al-Mallahi, T. Kataoka, H. Okamoto, An Algorithm for Distinguishing Potato Tubers on the Conveyor of the Potato Harvester using UV Camera. Proceedings of the ASABE Annual International Meeting-Providence, Rhode Island, USA, June 29-July 2, 2008, 084582.

[13] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, USA, 1992.

Table 1. Number of contact points detected using the developed Algorithm in clusters of different numbers of objects. Denominators show the total number of contact points and numerators show the number of contact points detected successfully.

Type of Contact Point	Number of Contact points in Similar Clusters			Total
	4-Object Cluster	3-Object Cluster	2-Object Cluster	
Potato-Potato (without clods)	141/141	100/100	65/65	306/306
Potato-Potato (with clods)	63(58)/63	11(11)/11	-	74(69)/74
Potato-Clod	96(90)/99	10(9)/10	3(3)/3	109(102)/112

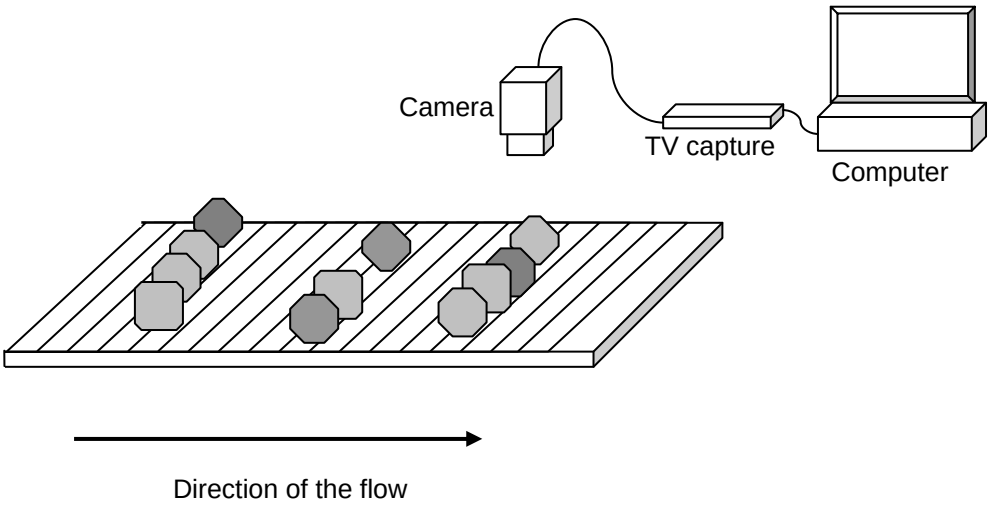

Figure 1. Schematic Diagram of the Image Capturing System

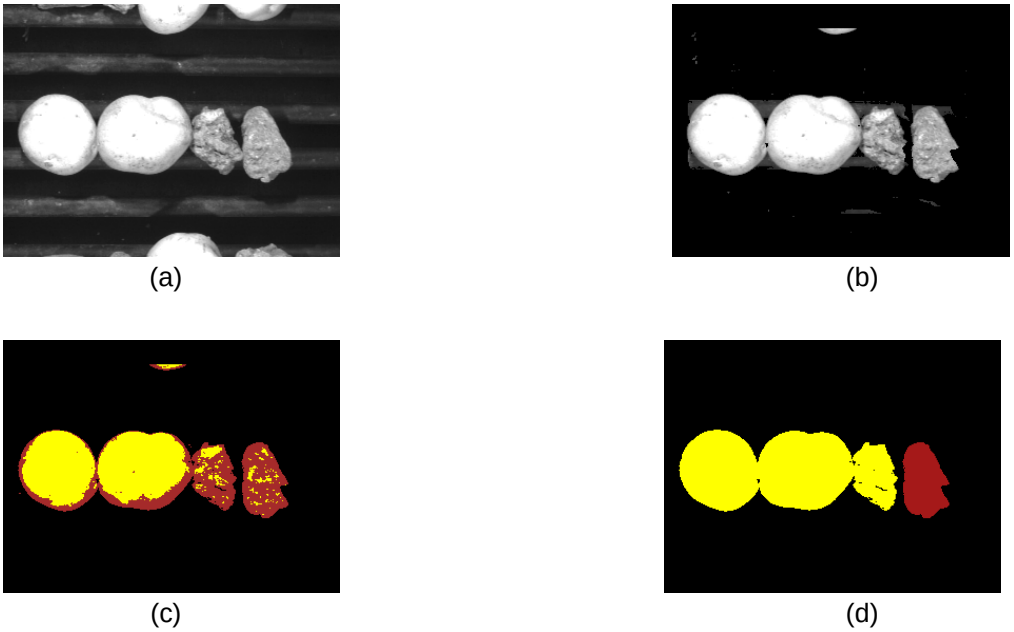

Figure 2. (a) Original image of 2 tubers and 2 clods arranged in-line. (b) Object-isolated image obtained after eliminating the background and the rods. (c) Pixel-segmented image obtained after classifying each pixel in the image to background (black), clod (red) or tuber (yellow) (d) Object-segmented image in which one clod was misclassified because it was attached to the neighbouring tuber

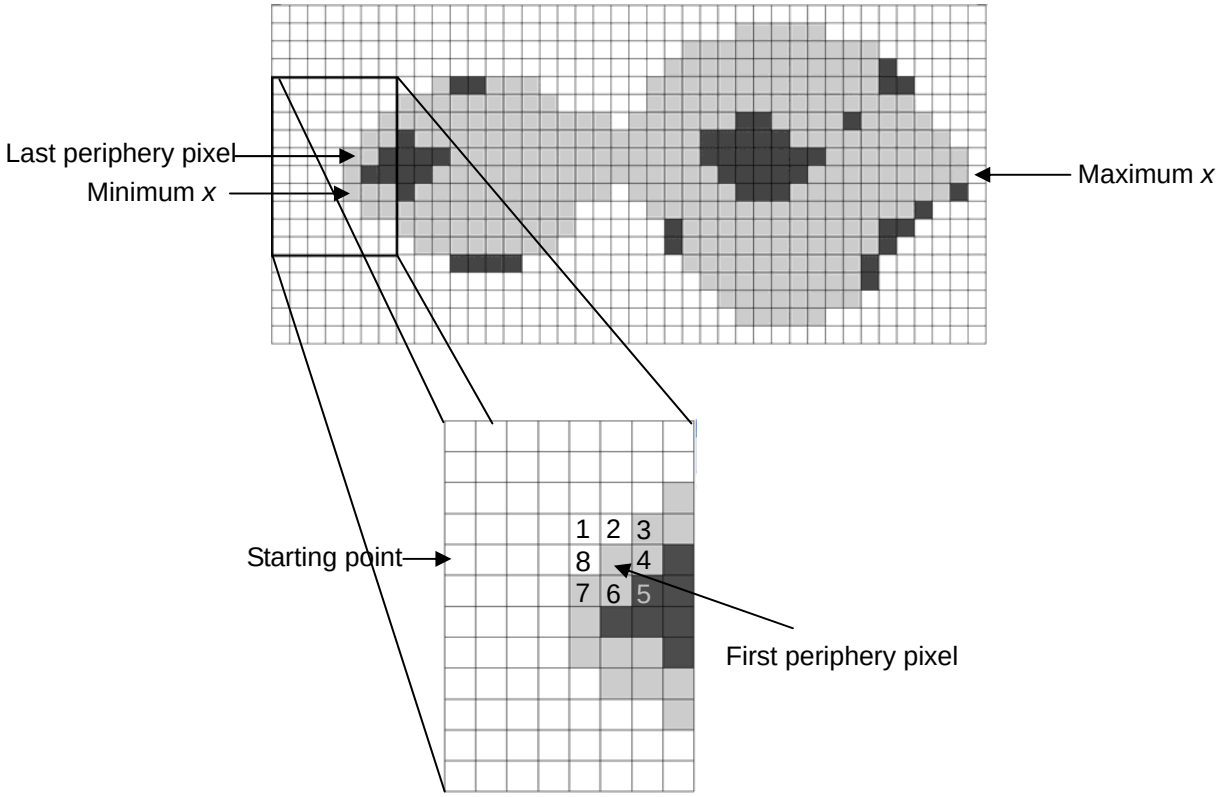

Figure 3. Simplified sketch of two adjacent potato tubers. White, light gray, dark gray pixels represent background, tuber and clod respectively. The magnified area shows the starting point for left-to-right scanning of the image. Once a periphery pixel is found a clockwise 8-neighbour scanning starts until finding the next periphery pixel which is number 3 in this example

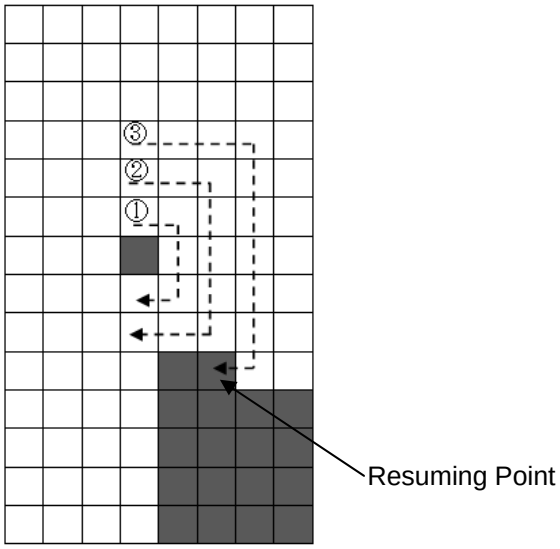

Figure 4. Progressive Scanning around the pixel of the minimum x-axis value until finding the next resuming point which is found in the third scanning in this example

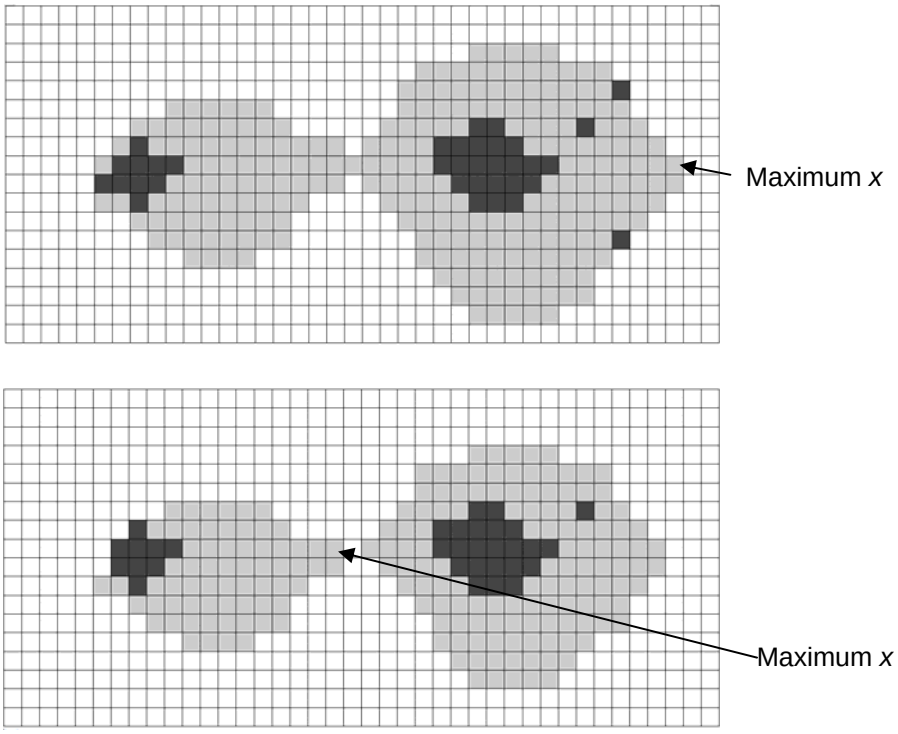

Figure 5. The process of shrinking the tubers in the sketch of Figure 3 by continuous contouring and erasing of the periphery until the contact point between the tubers is found. Once found, a vertical 1-pixel-thick black line is drawn in the original pixel-processed image.

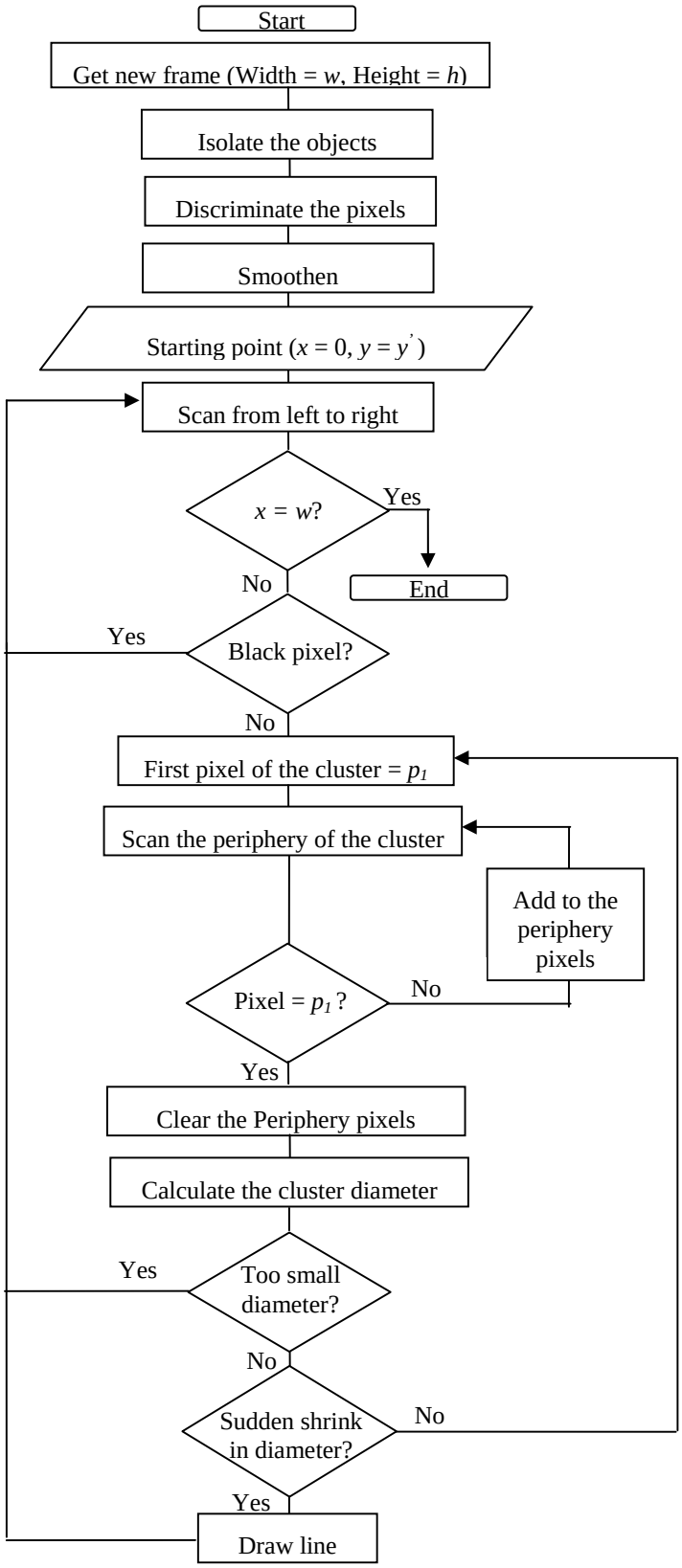

Figure 6. Flowchart of the process required to draw the lines at the touching points on the segmented video frame

(a) (b)

Figure 7. (a) Applying the developed algorithm on the pixel-segmented image (Figure 2 (c)) resulted in detecting the touching points (b) Object-segmented image after using the algorithm where all objects were detected correctly.