Restoration And Enhancement of Multi-Oriented Text Detection And Recognition For Natural Images

Karishma Jalal[1], R.B. Yadav[2]
[1, 2] Dept of Electronics and Communication Engineering
[1, 2] G. B. P. I. E. T, Pauri, Garhwal UK

Abstract- *Text detection has been a very challenging task for natural scene images as the text can be multi-oriented and can be of different fonts and sizes. Text detection is the process of extracting text from an image and text recognition is the process to convert the extracted text into an interpretable form. Our algorithm for text detection and recognition is based on SWT and MSER for detecting horizontal and near-horizontal oriented text. It finds application in a very wide range of areas such as navigation, aid visually impaired people. The main focus of this paper is to detect text particularly from sign board images.*

Keywords- Text detection, text recognition, SWT, MSER, segmentation, edge detection.

I. INTRODUCTION

Natural scene text is one of the general objects which one could frequently encounter on road signs, billboards, license plates of vehicles etc. Text contained in natural scene are a source of great information and detection and recognition of these text could drive a number of application such as visually impairment assistance, tourist assistance, content retrieval, ground vehicle navigation etc. Reading text from natural scene images is composed of two parts namely text detection and text recognition. Text detection is mainly the method of identifying or locating the text contained regions whereas the latter fundamentally deals with the interpretation of characters and symbols in the identified text region into machine readable form. Recognizing such text is a very challenging task owing to different orientations of text, variations in fonts, different characters, and interference from backgrounds.

II. METHODS AND MODELS

Many algorithms and methods have been proposed till now for the text detection and recognition from natural images like texture based method which treats text as a type of texture and variation in texture properties was used to distinguish text from image. This method worked well with the noisy, degraded images and images with complex background but has a disadvantage of being highly time consuming [13]. Second method is the Connected Component based method which is based on the detection of difference in the pixel value of the text and background and then grouping them to form connected region of text. It extracts the candidates from the image and then filters out candidates by manual rules or classifiers. It deals with the arrangement of the intensity values of an image. Methods such as detection using stroke width transform extraction of MSER (Maximally stable External Regions) regions fall under the connected component category. SWT and MSER are the most widely used detection algorithms because of their efficiency and stability.SWT algorithm makes use of canny edge to find edges of an image and from these responses calculate distance between two parallel edges. This helps to detect text line with similar stroke width and then filtering based on statistical properties of connected components is done to avoid the non text regions. MSER regions are connected areas characterized by almost uniform intensity, surrounded by almost uniform intensity, surrounded by contrasting background. They are constructed through a process of trying multiple thresholds, all pixels below a given threshold are white and all those above or equal are black. Selected regions are those that maintain unchanged shapes over a large set of thresholds i.e. the regions which show minimal variation with the threshold are defined as maximally stable. These MSER regions are then detected and filtered out to get the text contained region in the image.

Most of the state-of-the-art works in the field of text recognition in natural images are based on applying commercial OCRs over the bounding boxes where text has been detected. Commercial OCRs achieve an excellent performance when reading text of scanned images of documents, which are usually much contrasted images between the foreground and the background [1]. However, natural images present a further difficulty, because the text is usually embedded in complex backgrounds, the contrast between foreground and background is very low and the text can have many different appearances in terms of size, style, color and layout [1]. Some approaches have tried to face this problem, but most of them use their own datasets or are restricted to a limited number of font styles or even to only recognizing digits.

Segmentation is typically used in the detection of edges and boundaries of area of interest in an image. Image segmentation simply aims to change the representation of image into something which is easy to analyze and represent meaningful information of the particular area of interest. Image segmentation is the process of labeling each pixel of the image in such a way that pixel with same label is having two or more same characteristics on the basis of intensity value or any other characteristics of these pixels [10]. Image segmentation should be stopped at the point when object of interest or region of interest (ROI) has been detected. For example in an industry in which there is automated detection of electronic assemblies, interest lies on the fact of detecting specific anomalies such as missing component and broken paths . There is no point in carrying out segmentation beyond the level where the objective of segmentation is completed.

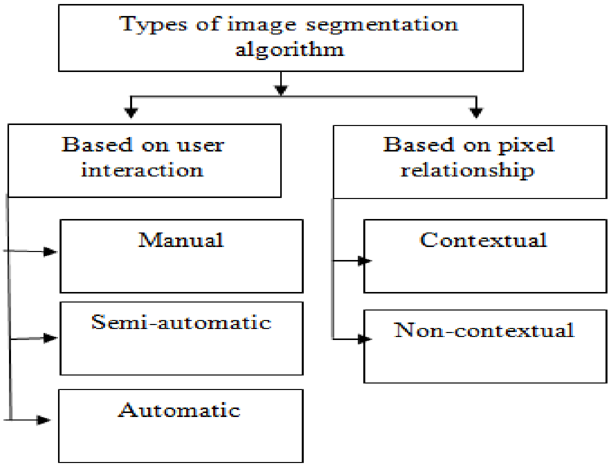

Types of image segmentation algorithm
Based on user interaction
Based on pixel relationship
Manual
Semi-automatic
Automatic
Contextual
Non-contextual

Figure 1: Figure showing basic classification of segmentation algorithm [13].

Based on user interaction, segmentation algorithm can be classified into the following categories:

1. Manual
2. Semi-automatic

III. METHODOLOGY

1. Input Image: - An RGB image of natural scene containing text is used as an input. The input image can be any natural scene image like billboards, license plates etc.

2. Pre- Processing – Image acquisition is a highly important step for quality control because it provides the input data for whole process, local illumination is directly linked with the quality of image acquisition because its variation can highly affect pattern visibility in image. Consequently, natural sources of light which are non-constant must not be employed and their influence should be completely eliminated this can be done with the help of pre- processing. There are various pre-processing methods such as:-

2.1. Contrast Adjustment- The contrast of an image is distribution of dark and bright pixels. A low contrast image exhibits small difference between its light and dark pixel values. Histogram of low contrast image is narrow. Human eye is sensitive to contrast rather than absolute pixel intensities, a perceptually better image could be obtained by stretching the histogram of an image so that full dynamic range of image is filled.

2.2. Intensity Adjustment- image enhancement techniques are used to improve an image. Intensity adjustment technique is a technique that maps image intensity values to a new range.

2.3. Histogram Equalisation – Histogram equalisation evenly distributes the occurrence of pixel intensities so that entire range of intensities can be covered. This method increases the global contrast of images, especially when usable data is represented by close contrast values. Through this adjustment the intensities can be better distributed on the histogram. This allows for the area of lower contrast to gain a higher contrast [10]. Histogram equalisation accomplishes this by effectively spreading out the most frequent intensity values. In more pronounced word we can say, histogram equalisation changes the pdf of given image into that of a uniform image that spread out from the lower pixel value (0) to a highest pixel value.

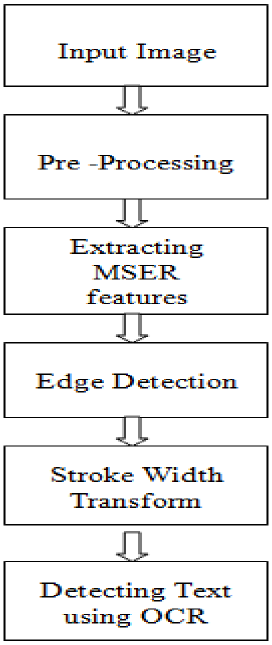


Figure2: Flow diagram for a methodology

Edges play a very important role in the processing of image. They provide the outline for the object of interest or region of interest (ROI). In the physical plane, edges correspond to the discontinuities in depth, surface orientation, change in material properties, and light variations. These variations are present in image as grayscale discontinuities. An edge is a set of connected pixels that lies on the boundary between two regions that differ in grey value. The pixels on an edge are called edge points. When edges are detected, the unnecessary details are removed, while only the important structural information is retained. Edges are significant intensity transitions.

An edge is extracted by computing derivative of the image function. It consist of two parts: magnitude of the derivative, direction of derivative, which represents contrast and orientation of edges respectively. Some of edges present in any image can be categorized as:

1. Step edges
2. Ramp edges
3. Spike edges
4. Roof edges

The idea is to detect the sharp changes in image brightness, which can capture the important events and properties. This is done in three stages:

1. Filtering: It is better to filter the image to get maximum performance for the edge detectors. This stage may be performed either explicitly or implicitly. It involves smoothing, where the noise is suppressed without affecting the true edges. It uses a filter to enhance the quality of edges in the image.

2. Differentiation: This phase distinguishes the edge pixels from other pixels. The idea of edge detection is to find the difference between two neighborhood pixels. If the pixels have the same value, the difference is zero. Detection of discontinuities can be done by calculating derivatives. Derivative used in this category is basically first order derivative and second order derivative.

3. Localization: In this stage edges are localized. The localization process involves determining the exact location of the edge. In addition, this stage involves edge thinning and edge linking steps to ensure that the edges is sharp and connected the sharp and the connected edges are then displayed.

The Prewitt operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. At each point in the image, the result of the Prewitt operator is either the corresponding gradient vector or the norm of this vector.

The Sobel operator, sometimes called the Sobel–Feldman operator or Sobel filter, is used in image processing and computer vision, particularly within edge detection algorithms where it creates an image emphasizing edges. The Sobel–Feldman operator is based on convolving the image with a small, separable, and integer-valued filter in the horizontal and vertical directions and is therefore relatively inexpensive in terms of computations. On the other hand, the gradient approximation that it produces is relatively crude, in particular for high-frequency variations in the image. The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images.

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise

2. Find the intensity gradients of the image

3. Apply non-maximum suppression to get rid of spurious response to edge detection

4. Apply double threshold to determine potential edges

5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

MSER stands for Maximally Stable External Regions. MSER regions are connected areas characterized by almost uniform intensity, surrounded by contrasting background. They are constructed through a process of trying multiple thresholds. The selected regions are those that maintain unchanged shapes over large sets of thresholds. The MSER algorithm has been used in text detection by Chen by combining MSER with Canny edges. Canny edges are used to help cope with the weakness of MSER to blur. MSER is first applied to the image in question to determine the character regions. To enhance the MSER regions any pixels outside the boundaries formed by Canny edges are removed. The separation of the later provided by the edges greatly increases the usability of MSER in the extraction of blurred text [8]. An alternative use of MSER in text detection is the work by Shi using a graph model. This method again applies MSER to the image to generate preliminary regions. These are then used to construct a graph model based on the position distance and color distance between each MSER, which is treated as a node. Next the nodes are separated into foreground and background using cost functions. One cost function is to relate the distance from the node to the foreground and background. The other penalizes nodes for being significantly different from its neighbor. When these are minimized the graph is then cut to separate the text nodes from the non-text nodes [9]. To enable text detection in a general scene, Neumann uses the MSER algorithm in a variety of projections. In addition to the grayscale intensity projection, he uses the red, blue, and green color channels to detect text regions that are color distinct but not necessarily distinct in grayscale intensity.

IV. RESULTS AND DISCUSSION

In the text detection and recognition, an RGB image of natural scene is considered as an input test image in MATLAB. The first conversion of RGB image is into greyscale and then it is converted into binary image. Then edge is detected using canny edge detection technique and MSER regions were detected using MATLAB. MSER regions were detected to differentiate between text and non-text regions. Then segmentation of image is calculated for removing unwanted detected regions. After the detection of only text part stroke width transform is performed to form the connected components from segmented region. At the last the image is masked for highlighting only text part from that image. Further GUI is implemented for better interaction with the user.

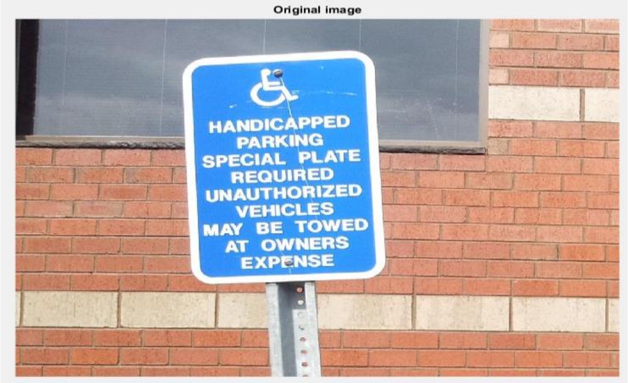


Figure3: The original input image for text detection

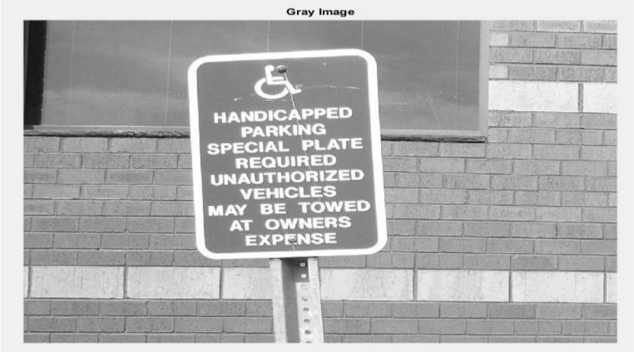


Figure 4: Conversion of original image into gray-scale image

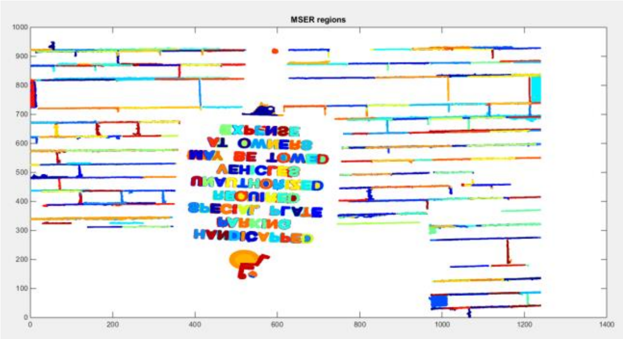


Figure 5: Image with MSER region detected

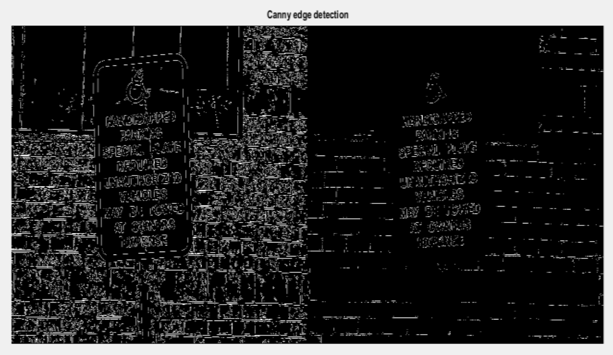


Figure 6: Canny edge detection of the original input image

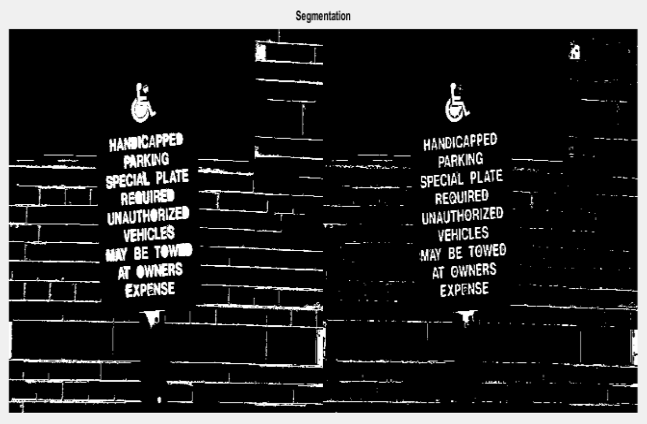


Figure 71: Segmentation is performed for tracing region of interest (ROI)

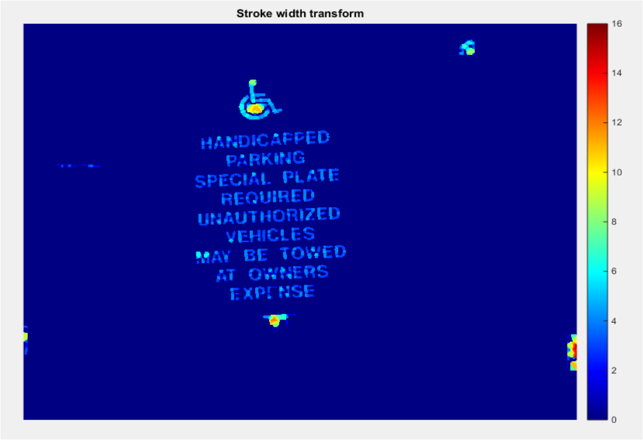


Figure 8: Stroke Width Transform of the image

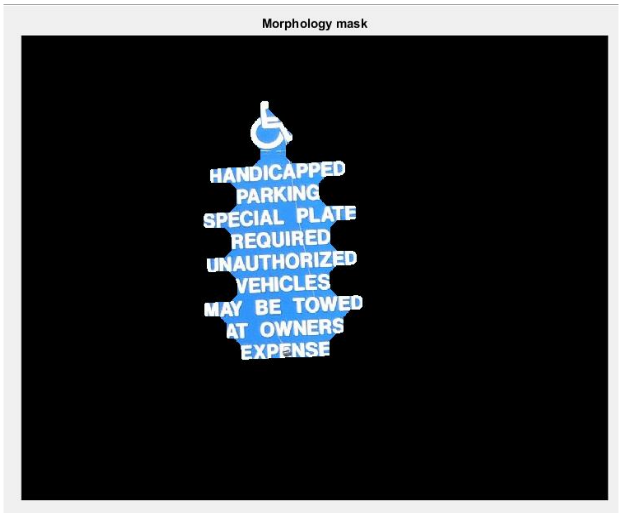


Figure 9: Morphology mask of the image

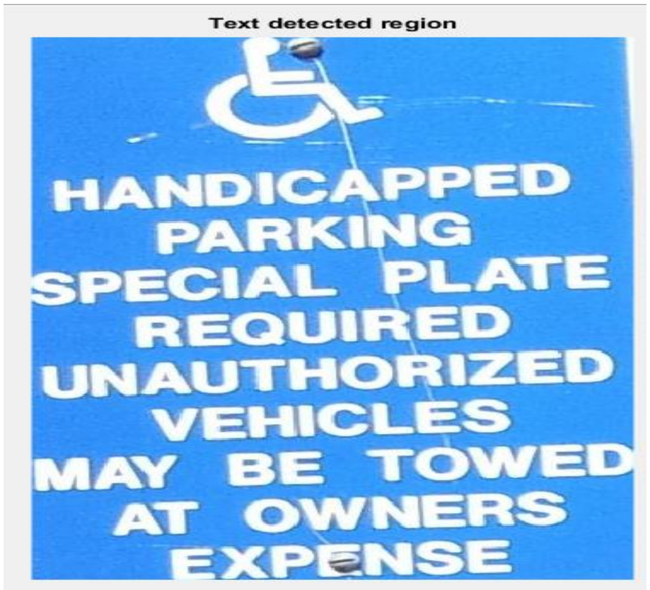


Figure 10: Image with only text detected regions

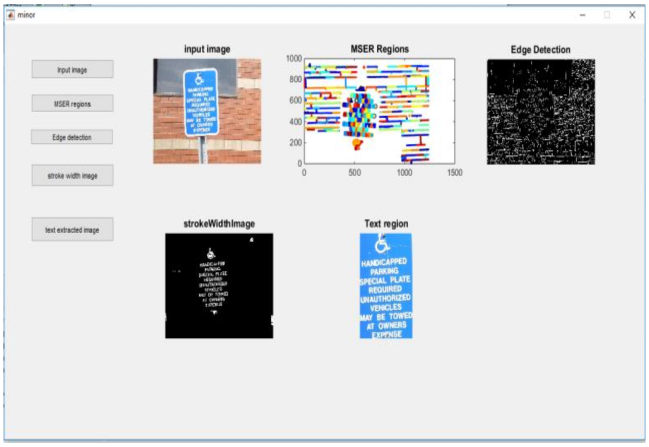
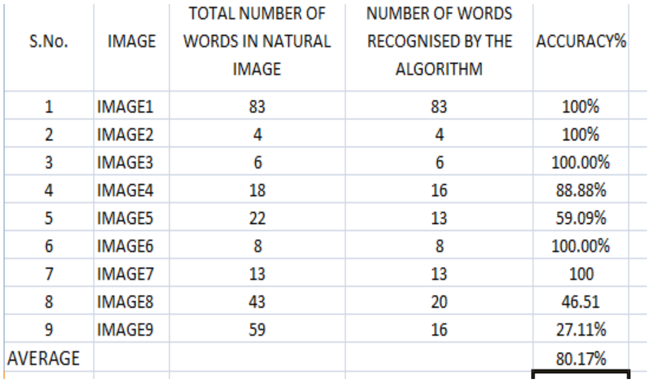


Figure 11: GUI for the proposed algorithm



S.No.	IMAGE	TOTAL NUMBER OF WORDS IN NATURAL IMAGE	NUMBER OF WORDS RECOGNISED BY THE ALGORITHM	ACCURACY%
1	IMAGE1	83	83	100%
2	IMAGE2	4	4	100%
3	IMAGE3	6	6	100.00%
4	IMAGE4	18	16	88.88%
5	IMAGE5	22	13	59.09%
6	IMAGE6	8	8	100.00%
7	IMAGE7	13	13	100
8	IMAGE8	43	20	46.51
9	IMAGE9	59	16	27.11%
AVERAGE				80.17%

Figure 12: Accuracy of proposed algorithm

V. CONCLUSION

In this paper, text detection and recognition from natural scene images has been done with a methodology based on Stroke Width Transform (SWT) and MSER (Maximally stable External Regions).This methodology was able to detect text from natural image with nearly horizontal text and text with arbitrary orientations. Here curled text lines are detected using the approach of connected components.

It can be further used for designing of a classifier based text detection and orientation which can recognize and detect multi-oriented text from natural scene images and then further conversion of the text detected into speech which could aid visually impaired people.

REFERENCES

[1] K. I. Kim, K. Jung, and J. H. Kim, "Texture-based approach for text detection in image using support vector machines and continuously adaptive mean shift algorithm," IEEE Trans.Pattern Anal. Mach. Intell., vol. 25, no. 12, pp. 1631–1639, Dec. 2003.

[2] X. Chen and A. L. Yuille, “Detecting and reading text in natural scenes,” in Proc. IEEE CVPR, Jun./Jul. 2004, pp. II-366–II

[3] T. E. de Campos, B. R. Babu, and M. Varma, “Character recognition in natural images,” in Proc. VISAPP, Feb. 2009, pp. 121–13

[4] B. Epshtein, E. Ofek, and Y. Wexler, “Detecting text from naturalscenes with stroke width transform,” in Proc. IEEE CVPR, Jun. 2010,pp. 2963–2970.

[5] C. Yi and Y. Tian, “Text string detection from natural scenes by structure-based partition and grouping,” IEEE Trans. Image Process., vol. 20, no. 9, pp. 2594–2605, Sep. 2011.

[6] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, “Detecting texts of arbitrary orientations in natural images,” in Proc. IEEE CVPR, Jun. 2012,pp. 1083–1090.

[7] F. Moosmann, E. Nowak, and F. Jurie, “Randomized clustering forests for image classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 9, pp. 1632–1646, Sep. 2008.

[8] X. C. Yin, X. Yin, K. Huang, and H. Hao, “Robust text detection in natural scene images,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36,no. 5, pp. 970–983, May 2014.

[9] V. Wu, R. Manmatha, and E. M. Riseman, “Textfinder: An automatic system to detect and recognize text in images,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 21, no. 11, pp. 1224–1229, Nov. 1999.

[10] Y. Zhong, K. Karu, and A. K. Jain, “Locating text in complex color images,” Pattern Recognit., vol. 28, no. 10, pp. 1523–1535, 1995

[11] X. Tong and D. A. Evans, “A statistical approach to automatic OCR error correction in context,” in Proc. 4th Workshop Very Large Corpora,1996, pp. 88–100.

[12] A. Ikica and P. Peer, “An improved edge profile based method for text detection in images of natural scenes,” in Proc. IEEE EUROCON, Apr. 2011, pp. 1–4.

[13] Cong Yao , Xiang Bai and Wenyu Liu, “ A unified framework for multi-oriented text detection and recognition” in Proc. IEEE Trans. Image Processing, May 2014.