Spatter Tracking in Laser- and Manual Arc Welding with Sensor-level Pre-processing

Olli Lahdenoja
Technology Research Center
Brahea Center, University of Turku, 20014 Turku Finland
olanla@utu.fi

Tero Säntti
Technology Research Center
Brahea Center, University of Turku, 20014 Turku Finland
teansa@utu.fi

Mika Laiho, Jonne Poikonen
Technology Research Center
Brahea Center, University of Turku, 20014 Turku Finland
mlaiho@utu.fi, jokapo@utu.fi

ABSTRACT

This paper presents methods for automated visual tracking of spatters in laser- and manual arc welding. Imaging of the welding process is challenging due to extreme conditions of high radiated light intensity variation. The formation and the number of spatters ejected in the welding process are dependent on the parameters of the welding process, and can potentially be used to tune the process towards better quality (either on-line of off-line). In our case, the spatter segmentation is based either on test on object elongatedness or Hough transform, which are applied on pre-processed image sequences captured by a high-speed smart camera. Part of the segmentation process (adaptive image capture and edge extraction) is performed on the camera, while the other parts of the algorithm are performed off-line in Matlab. However, our intention is to move the computation towards the camera or an attached FPGA.

Keywords

Laser welding, Thick Steel, Metal arc welding, Spatter tracking, Smart camera.

1. INTRODUCTION

One phenomenon in laser welding and welding in general is the formation of spatters, which represent the ejection of melt from the interaction zone caused by extreme conditions of material interaction. Increased spatter formation may indicate that the process parameters need to be adjusted to improve the quality.

In previous works [Nic11a], [Nic09a] the spatter formation was related to the on-line control of process parameters in welding of aluminum and 1mm thick steel sheets. In [Nic11a] the spatter segmentation was based on blob analysis including a threshold operation with a suitable mask. In [Jäg08a] a spatter tracking system incorporating a Kalman filter for trajectory analysis was proposed. In [Fen09a] spatter tracking in the context of laser MAG (metal active gas) welding was proposed. An off-line Kalman filter based approach for tracking was implemented with an SVM based spatter extraction.

In this paper, we present an approach for spatter tracking in the scenario of welding thick steel, by a combination of a smart camera for on-line processing and spatter event tracking as an off-line process, enabling automated analysis of the spatter behavior in welding. Different alternatives for spatter segmentation and tracking are considered in the context of high power laser welding and traditional manual metal-arc welding.

2. SPATTER EXTRACTION AND SEGMENTATION

On-line analysis

The huge difference between the intensity of light radiated from the welding zone compared to the surrounding areas causes a challenge for welding spatter segmentation and tracking. The objective is to enable the spatter object segmentation both within the very low and the very high intensity regions. This can be difficult to achieve because of the high intra-scene dynamic variation. Successful image capture may still be performed with the aid of active laser illumination [Fen09a]. Alternatively, a smart camera adapting to the intensity variation locally can be used.

Our approach is to use a high-speed smart camera, which enables the imaging of the welding zone by adjusting the integration time of the individual pixels according to their nearby local surroundings [Lah13a]. After this, the camera is used to extract the edge information from the second order spatial neighbors against an adjustable threshold. The KOVA1 array [KOV] provides a 96x96 resolution pixel parallel mixed-mode imaging processor, with a processor per pixel image processing functionality. In order to achieve a high enough frame rate for this application, only binary (1-bit) image sequence was used. Using gray scale images would limit the frame rate by filling up the bandwith of the 100 Mb/s Ethernet connection from the camera to the PC. Additionally, the required processing (edge detection) is performed faster on the camera than on the FPGA or the PC.

Off-line analysis

The spatter segmentation was performed as an off-line process in Matlab. Two slightly different methods for spatter segmentation are considered. The first is to apply Hough transform directly on the extracted edge images. This allows capturing the center location and perimeter of the spatters. The disadvantage of the method is that spatters which are not circular cannot be detected correctly. However, the method can also be implemented on FPGA for high-speed analysis.

The second considered approach is first to apply an additional morphological processing step to the extracted binary images (a morphological closing), in combination with connected component labeling. The closing operation could also be implemented on-line. The extracted object regions are then tested by their eccentricity (‘regionprops’ in Matlab R2012b). An additional step considered in this paper is to require that the center pixel of the object is of adjacent polarity compared to the edge pixels. This may be used to refine the circularity of the object, since it can be assumed that the spatters are, in general, symmetric and the edge image provided by the adaptive integration of the camera provides an object with a hole in the center.

3. SPATTER TRACKING

The proposed tracking algorithm is implemented in Matlab, and described in the following. First, the segmented objects are extracted as described in the previous section. A list containing the elements of initial spatter properties (SPL1) is created. Each spatter is assigned an individual ID number (in phase 3) and spatter area property, which describes a circular region in the case of Hough transform, and otherwise a rectangular region. TSPL_PRE (previous tracked spatter list) is initialized also to zero in this step. The steps of the tracking algorithm are explained below (1-5).

1) Initialize TSPL = 0 (Tracked spatter list) and read SPL2 (Spatter list 2) based segmented spatters at this frame.

2) Compare SPL2 to TSPL_PRE according to criterion IS_NEAR and update the matching spatters to TSPL. Remove these spatters from SPL2.

3) Compare the remaining SPL2 to SPL1, initialize and update those with property IS_NEAR_INIT to TSPL. Initialize spatter ID.

4) SPL1=SPL2, TSPL_PRE=TSPL

5) If frames left, Go back to phase 1

To initiate the tracking, the test IS_NEAR_INIT considers the spatter locations and areas only, without considering the previous displacement vector. In this phase, we use Euclidean maximum distance requirement of below 15 pixels. The test IS_NEAR performs the following action. The previous displacement vector (See Fig. 1) of the spatter under tracking has been stored and it is compared by using the IS_NEAR criteria to each of the new spatter observations. We use a Euclidean distance requirement of < 5 pixels. The assumption is that the spatters roughly propagate with constant velocity. An additional property which is checked is that the size of the new spatter has to be above 0.5 times and below 2 times of the size of the old spatter. The time which the spatter has been tracked (in number of frames) is also stored. All of the tracked spatters are collected to a list for later analysis. The post-processing step incorporates also a possibility to consider the average sizes and velocities of the spatters under test.

Observe that we use the same tracking algorithm in both cases i.e. with the connected component labeling based segmentation and with the Hough transform based segmentation. Thus, the segmentation algorithm, which could be performed on the camera or on an FPGA, is not affected by the tracking algorithm. Further possibilities of HW accelerating the tracking algorithm are discussed in section 6.

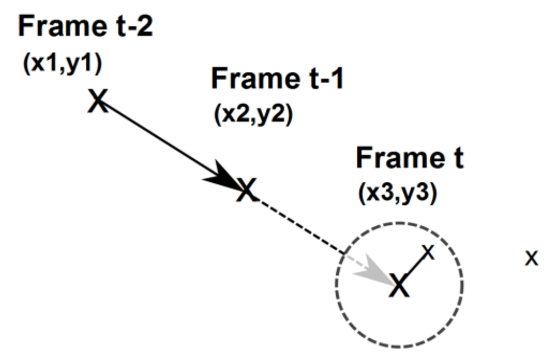

Figure 1. The test for proximity between two spatters. The estimated new center location of the spatter according to previous frames is compared to each of the candidate spatters in Frame t according to Euclidean distance (partly reminding the Kalman filter [Jäg08a]).

4. EXPERIMENTAL SETUP

Manual metal-arc welding set-up

The first tests were performed with manual arc welding, where the number of spatters is very large. The experimental setup in this case was initially targeted to test the properties of the camera, but the excessive spatter formation was shown to be suitable also for testing the spatter tracking algorithm. The frame rate used was 1408 Fr/s (with a difference between two frames of approximately 710us). Fig. 2 represents an example of the adaptive camera capture and the corresponding edge image captured on-line.

Laser welding set-up

The second test case used was a coaxial setup of full-penetration mode laser welding of 4mm thick steel. The laser power applied was selected so that full penetration was achieved, 4.5kW on 12mm/s and 6kW on 20mm/s welding speeds. The number of spatters was very low, which created challenges for the automated spatter analysis due to a relatively large portion of vapor plume. A partial penetration mode is expected to produce more spatters, which will be studied in the future. The frame-rate applied was approximately 3500 Fr/s.

5. EXPERIMENTS

Manual metal-arc welding tests

Hough transform was found to suite well to the case of manual metal arc welding, since the spatters were typically circular (See Fig. 2). The considered radiuses were from 2 to 10, with a Hough sensitivity parameter (HS) of 0.85 (the available implementation in Matlab R2012b). However, large spatters (e.g. r>6) were very rare, due to larger imaging area in comparison with the laser welding. Connected component labeling, which was applied after a binary closing with a 3x3 mask of all ones was used as a reference method. The object elongatedness was required in this case to be below 0.9.

Since it was not possible to manually label the whole welding sequence, the performance of the tracking was estimated so that a ground truth was generated and the overall spatter count returned by the algorithm (i.e. TP+FP) was determined at constant 100 frames steps (see Fig. 4, index 1 indicates frames 1-100 etc.). Note also that the ground truth contains all spatters, but the tracking algorithm only counts spatters included in at least 3 succeeding frames.

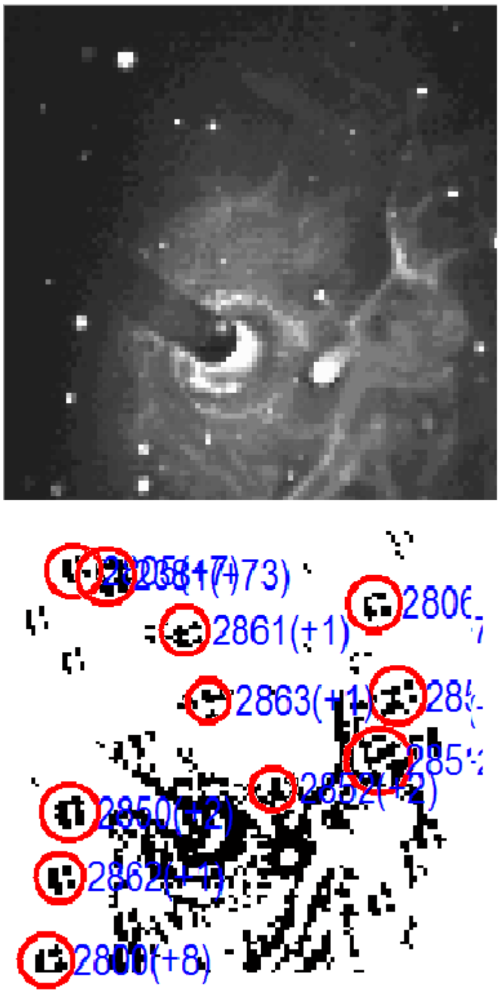

Figure 2. The arc welding set-up. The image on the top represents the welding zone captured by the adaptive integration. On the bottom the edges extracted on-line and the tracked spatter IDs extracted off-line.

 ISBN 978-80-86943-72-5

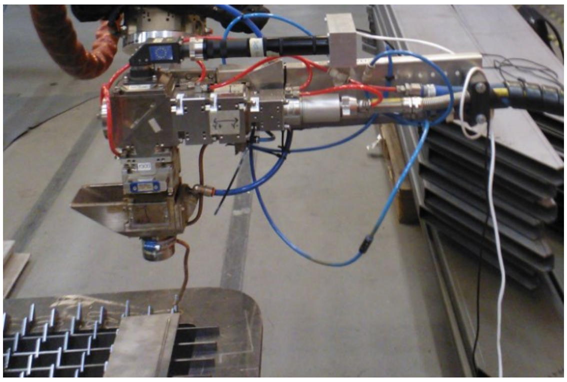

Figure 3. The laser welding test setup. The camera is attached to the welding head (from Precitec) through a 90 degree mirror and focusing lens. The fibre laser is brought from a collimator through 90 degree mirror optics to the work piece. The focal position of the laser was -4mm (below the surface).

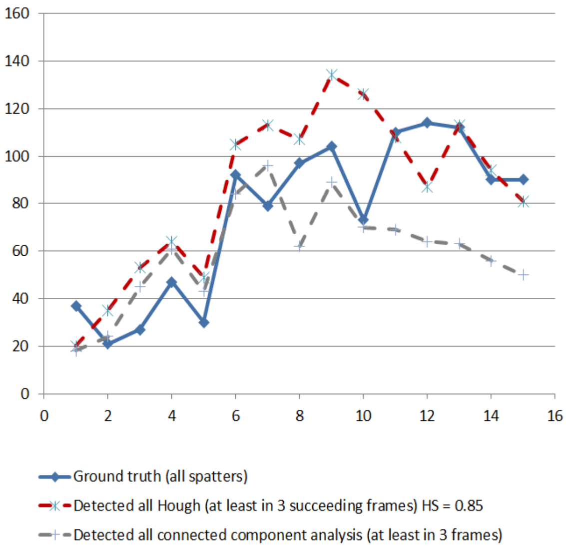

Figure 4. An indicative chart on the ground truth and the overall number of detected spatters in arc welding in frames 1-1500 at 100 frame intervals.

In the tests, increasing the requirement of the number of consecutive tracked frames could be used to filter out false positives caused by the vapor plume and to make it possible to increase the sensitivity of the segmentation. Also, a rectangular area near to the arc end was masked to reduce the effect of the plume. The ground truth was generated by three human experts in 500 frame intervals, which was quite challenging and could induce certain bias. It can be observed, that results follows the manually extracted spatter behavior. It appears in Fig.4, that when the number of spatters (and likely the amount of plume) is higher, the Hough transform gives better results. In Fig. 5, an angular histogram plot of the spatter directions for the same sequence is shown with Hough transform (3 succeeding tracked frames required).

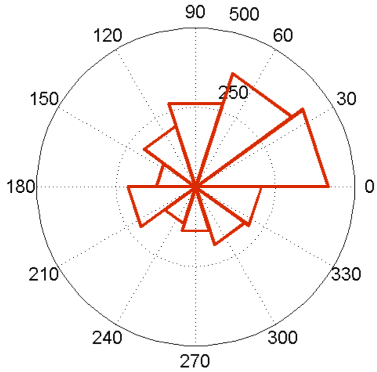

Figure 5. An angular histogram plot of the most common spatter directions in the arc welding test using Hough transform. The location of the arc was approximately in the location shown in Fig. 2.

Laser welding tests

Since the spatters in laser welding were not always circular, the connected component analysis in combination with spatter elongatedness test was applied with the same parameters as in arc welding. An additional requirement was that the center location of the spatters was white, which was shown to reduce the false alarm rate. The size of the detected spatters was limited to a square of minimum edge dimension of 5 pixels and maximum edge dimension of 20 pixels.

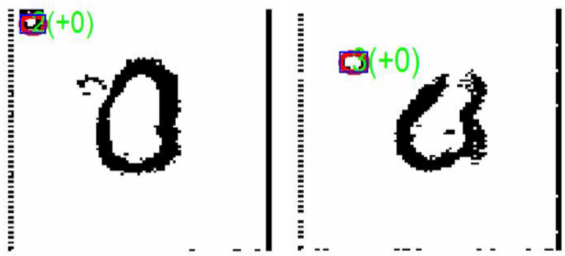

Figure 6. Examples of tracked spatters in laser welding scenario.

A ground truth of first 1000 frames of the sequence was manually collected, including only 13 spatters. The total amount of spatters was very low due to the used full penetration mode. A total of 9 spatters were detected correctly in comparison with the ground truth with one false alarm. Thus, a detection rate of 69.2% was obtained with false positive rate of 0.001/frame. Fig. 6 represents examples of tracked spatters in the laser welding scenario.

In [Nic11a] the false positive rates of spatter segmentation varied between 0.009/frame and 0.028/frame. The detection rate is not directly comparable, since the authors in [Nic11a] consider the detection of spatter events as an occurrence of one or more spatter within a single image. If there were one or more spatters present in the ground truth and also in the segmentation result, the match was considered as positive. In our case, individual spatters, which may locate across multiple succeeding frames are considered. However, it should be noted, that the tracking option used in our paper relaxes the requirement for spatter counting, since if a spatter is not found from an individual frame, the correct detection can still be attained in the succeeding frames.

6. ON POSSIBILITIES OF HW ACCELERATING THE ALGORITHMS

We are currently studying options for implementing the segmentation and the tracking algorithm on FPGA or by in-camera processing. While examining seam tracking, we have already demonstrated Hough transform based line detection on the FGPA attached to the camera [Sän14a]. The unit will be modified to detect circles, and then used as the detection engine for the tracking algorithm presented in this paper. Also the FPGA based acceleration of the tracking algorithm is possible. Currently on laptop PC with Matlab the algorithm runs approximately 30fps (without considering the time for the segmentation).

FPGA based processing

In order to implement the tracking algorithm on an FPGA the number of segmented spatters (elements in the lists) sets the limit for the efficient implementation. This is because the parameters of the spatter elements (e.g. 8-bit values) should be, preferably, mapped to the internal FPGA Block RAMs and not to external memory, which introduces extra latency. Assuming a BRAM size of 36kbit and list element size of 6*8 = 48 bits, the maximum number of elements in a list can be estimated. When mapped to a single BRAM the maximum length of a list would be 768, which should be enough for the application considered. The collection of the final spatter properties for further analysis could then be implemented by sending the data to a PC. At this point, as mentioned, only the adaptive image capture and spatter edge extraction were performed on-line.

Camera based processing

The applied smart camera (KOVA1) includes pixel-level processing functionality based on Cellular Nonlinear Network (CNN)-type [Ros93a] parallel processing architecture. In the KOVA1 CMOS sensor chip, analog and digital (mixed-mode) computational circuitry, as well as binary (static) and analog (dynamic) memory is integrated physically together with each sensor diode, creating a focal plane cell [Lai11a]. The additional pixel-level circuitry naturally imposes some limits on practical image resolution vs. chip size (cost/yield), e.g. the size of the commercial KOVA1 sensor-processor array is 96x96 pixels, while the largest published research implementation with a similar cellular processing architecture is the SCAMP-5 with a 256x256 pixel array [Car13a].

Each pixel cell in the sensor array of the KOVA1 is directly connected to its nearest neighborhood and in some operations even to the second neighborhood. Information propagation over larger distances is also possible in an asynchronous or synchronous manner. The cellular connectivity together with local, pixel-level, memory and processing circuitry allows the implementation of multiple-stage image analysis operations on the pixel plane. The pixel-level parallel processing and the avoidance of data transfer to an external processor result in reduced energy consumption and more importantly considering the present application, greatly reduced processing delay. The pixel-level processing functionality of the KOVA1 includes both grayscale and binary (1 bit) operations. The grayscale processing was used in this case to implement the image dynamic compression used in the capture of the welding data. The preprocessed grayscale image is segmented into a 1 bit binary representation to allow more efficient pixel-level shape and object analysis.

At this stage an edge detection operation has been used to extract relevant detail. The pixel level processing circuitry enables the implementation of sensor-level binary mathematical morphology and other similar analysis, as described in [Lai08a], as well as normal Boolean logic operations. By storing and combining intermediate result images in local pixel memory even complex spatial processing sequences can be performed on the sensor plane, including time-domain inter-frame analysis. Asynchronous binary propagation on the sensor plane can be used to perform regional operations such as holefilling, or morphological reconstruction very efficiently. By applying more binary analysis on the sensor-level, the processing demands at later stages (e.g. on an FPGA) can be greatly reduced, while retaining sufficient overall analysis speed.

If edge image based spatter segmentation is applied, one possibility to perform more computation at the camera level would be to use a holefiller template with pixel-level processing to detect roundish spatter objects, and then to extract the center locations of these objects for further processing. However, the holefiller template requires, that the edges of the objects are fully connected, which may not be always achievable. This could be avoided by using orientation selective propagation templates and by combining them or by applying a combination of dilation and skeleton extraction. Also blob based region extraction could be considered in the future.

7. DISCUSSION

The test set used in laser welding was challenging, since the spatters were not always circular, and due to second order neighborhood used in smart camera based segmentation. A second order neighborhood was used to better obtain the smooth edges of the spatters. Due to this, the edges of the spatters were thick, thus reducing the freedom in applying a larger closing mask. The purpose of the closing mask is to avoid a situation, where an individual spatter is detected as two distinct objects in the connected component labeling step. Also, if using too large closing mask, the centroid of the spatters becomes filled, and the segmentation process becomes less robust.

An approach for spatter segmentation and tracking in two welding scenarios was proposed. Our approach extends the work in [Nic11a] in that it also enables the spatter tracking and the analysis of the individual spatter properties. Future work consists of developing methods towards on-line spatter analysis. In comparison with work in [Lah13a] the methods proposed provide further robustness to false positives induced by the vapor plume. A limitation of the proposed approach in comparison with traditional multi object tracking is that the collision of two spatters may lead to erroneous tracking. This issue should be addressed in the future.

8. CONCLUSIONS

Methods for spatter segmentation and tracking in two welding scenarios were proposed in this paper. Hough transform was proposed for extracting spatters in the manual-arc welding scenario, while object elongatedness analysis was proposed for laser welding. The advantage of the Hough transform based segmentation is that it inherently avoids merging two distinct objects (or plume) nearby into the same object in the segmentation phase. The main advantage of the connected component analysis approach and the elongatedness test was that the objects need not to be circular, allowing flexibility for segmentation. Checking the polarity of the center pixel of the extracted object also increased the robustness of the segmentation algorithm.

9. ACKNOWLEDGMENTS

We would like to acknowledge the support from Lappeenranta University of Technology (LUT) and Machine Technology Center Turku in conducting the tests. The research was funded by Academy of Finland project no. 254430.

10. REFERENCES

[Nic11a] Nicolosi, L., Tetzlaff, R, Blug, A. , et.al, A Monitoring System for Laser Beam Welding Based on an Algorithm for Spatter Detection. Conf.proc. ECCTD'11, pp. 25-28.

[Nic09a] Nicolosi, L, Tetzlaff, R., F. Abt, et. al, New CNN based algorithms for the full penetration hole extraction in laser welding processes: Experimental results, IJCNN'09, pp. 2256-2263.

[Jäg08a] Jäger,. M, Humbert, S., Harmrecht, F. A. , Sputter Tracking for the Automatic Monitoring of Industrial Laser-Welding Processes. IEEE Trans. Industrial Electronics, 55, No.5, 2008.

[Fen09a] Fennander, H., Kyrki, V., Fellman, A., Salminen, A., Kälviäinen, H., Visual measurement and tracking in laser hybrid welding, Machine Vision and Applications 20, pp. 103–118, (2009).

[Lah13a] Lahdenoja, O., Säntti, T, Poikonen, J., Laiho, M., Paasio, A., Characterizing Spatters in Laser Welding of Thick Steel Using Motion Flow Analysis, SCIA'13, pp. 675-686.

[KOV] http://www.kovilta.fi

[Sän14a] Säntti, T., Lahdenoja, O., Paasio, A., Laiho, M., Poikonen, J., Line Detection on FPGA with parallel sensor-level segmentation, CNNA'14, to Appear.

[Ros93a] Roska, T., Chua, L. O., The CNN Universal Machine: An Analog Array Computer, IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing. 40(3), pp. 163-173, (1993)

[Lai11a] Laiho, M., Poikonen J., Paasio A.: MIPA4k: Mixed-Mode Cellular Processor Array. In: Zarandy A. (ed.): Focal-Plane Sensor-Processor Chips, Springer (2011)

[Car13a] Carey, S.J., Lopich, A., Barr, D.R.W., Wang, Bin, Dudek, P., A 100,000 fps vision sensor with embedded 535GOPS/W 256×256 SIMD processor array, Symposium on VLSI Circuits (VLSIC), pp. 182-183, (2013)

[Lai08a] Laiho, M., Paasio, A., Flak, J., et. al, Template Design for Cellular Nonlinear Networks With 1-Bit Weights. IEEE Trans. on Circuits and Systems 55(3), pp. 904-913 (2008)