Brightness Based Selection and Edge Detection Based Enhancement Separation Algorithm for Low Resolution Metal Transfer Images

Zhen Zhou Wang and YuMing Zhang, Senior Member, IEEE

Abstract—Next generation gas metal arc welding (GAMW) machines require the rapid metal transfer process be accurately monitored to achieve high speed and control. However, the necessity for high frame rate reduces the resolution achievable and bright welding arc makes it difficult to clearly image the metal transfer process. Processing of images for real-time monitoring of metal transfer process is thus challenging. To address this challenge, the authors analyzed the characteristics of metal transfer images in a novel modification of GAMW, referred to as double-electrode GMAW, and proposed an algorithm consisting of a system of effective steps to extract the needed droplet feedback information from high frame-rate low-resolution metal transfer images. Experimental results verified the effectiveness of the proposed algorithm in automatically locating the droplet and computing the droplet size with an adequate accuracy.

Note to Practitioners—Monitoring of metal transfer process is a fundamental step toward intelligent control of gas metal arc welding process and its modifications. However, the metal transfer rate may exceed over hundred hertz and its monitoring requires high frame rate images so that the resolution of the image is relatively low. Because of the low resolution and harsh welding environment, automated processing of the images for droplet identification and computation is challenging. This paper proposes a system of solutions to process the low resolution images to obtain robust and accurate estimation of the droplet location and size. Experiments verified the effectiveness of the proposed solutions and future work will focus on algorithm optimization and high speed processor implementation to improve the speed for real-time control of droplet trajectory and size which are required for future precision manufacturing applications.

Index Terms—Edge detection, GMAW, Image Processing, Interpolation, Machine Vision, Metal Transfer, Welding.

I. INTRODUCTION

In GMAW, the arc is established between the work-piece and a continuously-feeding wire which is melted primarily by the anode heat. The anode heat is proportional to the current and in a typical operation the power supply automatically adjusts the current to maintain the arc voltage at the pre-set value. Because the arc voltage is a measurement of the arc length (distance from the tip of the wire to the work-piece), the arc length is maintained at the desired value and the balance between the feeding and melting is thus maintained.

Because of the consumable electrode (wire), GMAW has a higher productivity than another widely used arc welding process—gas tungsten arc welding or GTAW—that uses an un-consumable tungsten electrode to establish an arc to the work-piece. Although the consumable electrode increases the productivity, the melting of the wire and the transfer of the molten metal from the wire to the work-piece complicate the process and reduce the arc stability and regularity. In particular, the way how the molten metal is transferred into the weld pool or the metal transfer mode plays an important role in determining the arc stability and weld quality. It is preferred that the droplet of the molten metal be transferred into the weld pool in particles which is smaller than the diameter of the wire. If possible, it is desired that the droplets are detached and controlled with desired uniform diameters at desired frequency. Machine vision appears to be the only way which may provide accurate feedback to facilitate such a control.

Double-electrode GMAW or DE-GMAW [8] is a novel modification of traditional GMAW which adds a tungsten electrode to bypass part of the current that melts the wire from the work-piece (Fig. 1). As a result, the current is decomposed into the bypass current and the base metal current. This decomposition provides a novel way to reduce the base metal current thus the base metal heat input without reducing the melting speed of the wire. Further, it is found that the trajectory and diameter of the droplets can be effectively adjusted by the current distribution. As a result, DE-GMAW provides a means to accurately and better control the weld bead profile through the control of the diameter, trajectory, and frequency of the droplets and the base metal heat input and to accurately and better control the weld bead metallurgy for precision welding, brazing, and rapid prototyping. However, to facilitate a feedback control, its
metal transfer process must be monitored through image processing.

To understand and study the metal transfer process in GMAW, high speed cameras have been extensively used to image and record the dynamic developments of the droplets for off-line manual analysis. However, the control of metal transfer using image as feedback which requires automated, robust, and high speed image processing has not been explored. Because of the relative high frequency of metal transfer, the camera must be operated at relatively low resolution to image the metal transfer process with an adequate frequency. In addition, the bright welding arc makes it difficult to clearly image the metal transfer process. Automated detection of the droplets from low resolution metal transfer images is thus challenging but is a fundamental issue which must be resolved toward the developments of next generation machines for GMAW and its modifications including DE-GMAW where metal transfer plays a critical role in determining the quality of resultant welds.

II. PROBLEM DESCRIPTION

Fig. 1 also shows a typical low resolution metal transfer image in DE-GMAW captured with a frame rate 3000 frames/sec. At such a high frame rate when the data transfer rate is fixed, only part of the image can be used. The resolution of the acquired image is $260 \times 360$ for the camera used, OLYMPUS i-SPEED which is considered among those with highest data rate and can capture images at up to 33,000 frames per second. Furthermore, the practical distance between the camera and welding arc must be relatively far. Hence, the objects of interest are only in a small area in the image.

III. BRIGHTNESS BASED SELECTION AND EDGE DETECTION BASED ENHANCEMENT SEPARATION

First, the bilinear interpolation is adopted to increase the resolution. The bilinear interpolation for point $(x, y)$ to be interpolated is performed using

$$f(x, y) = ax + by + cxy + d \quad (1)$$

where the coefficients $a, b, c$ and $d$ are obtained using the four neighbors as:

$$\begin{bmatrix}
    a \\
    b \\
    c \\
    d
\end{bmatrix} = \begin{bmatrix}
    x-1 & y-1 & (x-1)(y-1) & 1 \\
    x+1 & y-1 & (x+1)(y-1) & 1 \\
    x-1 & y+1 & (x-1)(y+1) & 1 \\
    x+1 & y+1 & (x+1)(y+1) & 1
\end{bmatrix}^{-1} \begin{bmatrix}
    f(x-1,y-1) \\
    f(x+1,y-1) \\
    f(x-1,y+1) \\
    f(x+1,y+1)
\end{bmatrix}$$

Once the bilinear interpolation is done for all possible points in the area of interest, the resolution will be doubled. The bilinear interpolation can then be used again on the image which has been interpolated to further increase the resolution. In this paper, the authors chose to increase the resolution five times. Figs. 2 and 3 show part of an original metal transfer image and its resultant image after the bilinear interpolation. Edges detected from them using the SOBEL mask are shown in Fig. 4. It is apparent that much more detailed information about the edge has been detected from the interpolated image.

In the image shown in Fig. 1, the central part is the welding arc, on which a bright droplet can be seen. The image processing algorithm needs to locate the droplet/droplets in the image and calculate its/their diameter/diameters. For sequential images, the trajectory and transfer rate (number of droplets per second) of the droplets will be able to be computed accordingly.

Second, the authors use the minimum error thresholding method to determine a threshold $T$ whenever segmenting a two part (darker and brighter) image:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln \left( \frac{P_2}{P_1} \right) \quad (3)$$
where \( \mu_1 \) and \( \mu_2 \) are the mean grayness for the darker and brighter part respectively, \( \sigma^2 \) is the variance of the total grayness. \( P_1 \) and \( P_2 \) are the probabilities of the dark and bright part respectively, which are specified in advance.

Third, the proposed brightness based selection and edge based separation algorithm makes use of the reality that part of the droplet’s brightness is higher than other regions. Based on the brightness, the image of the arc area can be divided into several areas, depending on how many droplets the image contains. Only the areas that contain a droplet are processed in order to reduce the processing time. These specific areas are defined as ROIs (regions of interest) in this paper. However, to overcome the difficulty caused by the fact that the droplet is not uniformly brighter than other regions, the proposed algorithm will also make use of the droplet edge.

The proposed algorithm can be described as ten steps.

**Step1: Preprocessing**

This step is to separate the objects of interest completely from the background which is completely dark. To this end, the image is binarized:

\[
f'(x, y) = \begin{cases} 
1, & \text{if } f(x, y) > T \\
0, & \text{if } f(x, y) < T 
\end{cases}
\]  

where the threshold \( T \) is determined using Eq. (3). Fig. 5 shows an original image in (a) and its binarized image in (b). The binarized image is then filtered to eliminate unwanted noises to ease the computation of the original position of the droplet. In this step, the largest bright area is selected as \( S \) and then the filtered image \( \hat{f}(x, y) \) is determined as:

\[
\hat{f}(x, y) = \begin{cases} 
f'(x, y), & \text{if } (x, y) \in S \\
0, & \text{if } (x, y) \notin S 
\end{cases}
\]  

The filtered image is shown in Fig. 5(c).

**Step 2: Identification of the wire tip**

The droplet is formed at and detached from the tip of the wire. In the image, it is located at the top of the welding arc. To identify it, the coordinates of the topmost point of the welding arc needs to be calculated. Since the image has been binarized, the computation appears straightforward. Its position, denoted as \( \hat{p} \), can be computed using the equation:

\[
\hat{p} = (x, \min(y)), \text{ for } (\hat{f}(x, y-1) = 0) \& (\hat{f}(x, y) = 1)
\]  

The computed coordinate for the image in Fig. 1 is \((122, 224)\). Since the arc almost remains constant during the welding, this coordinate can be used for the whole video sequence.

**Step 3: Binarization for droplet identification**

In this step, the original image is re-binaried, but for the arc area, toward the identification of the droplet. To this end, Eq.4 is used again but with a higher new threshold which is determined also using Eq. (3) but from the histogram of the arc area. Fig. 6 shows the resultant binarized image.

**Step 4: Image filtering**

Because multiple droplets may exist and the number of the droplets is unknown, a pre-specified threshold of area is used to determine if a particular bright area in the binarized image is a droplet or noise. The union of all bright areas which are larger than the threshold is defined as \( S \) and Eq. 5 is then used to filter the image. The filtering result for Fig. 6 is demonstrated in Fig. 7.

**Step 5: Selection of ROI based on the result in Step 4**
Once the binarized image is filtered, the areas binarized as 1 represent where the droplets are located. For example, if there are two droplets in the metal transfer image, then there will be two main areas with value 1 after binarization. Each ROI is defined based on the center of each area with value 1. The following equation is used to compute the center of these areas.

\[ x_c = \frac{x_{\text{max}} + x_{\text{min}}}{2}, \quad y_c = \frac{y_{\text{max}} + y_{\text{min}}}{2} \]  

After the centers are found, the ROIs are defined as 20×20 squares around corresponding centers, where the window 20×20 is selected based on the observation of typical diameter of droplet which is not greater than 10 pixels.

**Step 6: Bilinear interpolation of the selected ROI**

After all the ROIs are defined, each of them is interpolated by Eq.1 for a five-time increase of resolution.

**Step 7: Edge detection of interpolated high resolution ROI**

The following four SOBEL operators are used to find the edges:

\[
S_x = \begin{pmatrix}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1
\end{pmatrix}, \quad S_y = \begin{pmatrix}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -2 & -1
\end{pmatrix},
\]

\[
S_d = \begin{pmatrix}
-2 & -1 & 0 \\
-1 & 0 & 1 \\
0 & 1 & 2
\end{pmatrix}, \quad S_{id} = \begin{pmatrix}
0 & 1 & 2 \\
-1 & 0 & 1 \\
-2 & -1 & 0
\end{pmatrix},
\]

where \( S_x \) is used to find the edge in x direction, \( S_y \) in y direction, \( S_d \) in diagonal direction and \( S_{id} \) in inverse diagonal direction.

Fig. 4(b) illustrates the result of SOBEL edge detection from an interpolated image.

**Step 8: K-means clustering of edge information**

Since all edges in the ROI will be detected, each ROI may contain some edges that may not belong to the droplet and should be eliminated. An example ROI is shown in Fig. 8, it is seen that the bottom-left regions contain edges which do not belong to droplet this ROI corresponds to.

Every droplet’s edge is defined as a pattern class and the prototype of each pattern class is the mean vector of the patterns of that class:

\[ m_j = \frac{1}{N_j} \sum_{i \in w_j} x_j \]  

where \( N_j \) is the number of pattern vectors from class \( w_j \). One way to determine the class membership of an unknown pattern vector \( x \) is to assign it to the class of its closest prototype. Using the Euclidean distance to determine closeness reduces the problem to computing the distance measures:

\[ D_j(x) = \| x - m_j \| \]  

where \( \| \| \) is the Euclidean norm. \( x \) is assigned to class \( w_i \) if \( D_j(x) \) is the smallest distance. That is, the smallest distance implies the best match in this formulation.

By computing the distance, each droplet’s edge can be separated. The class with the largest \( N_j \) which contains most edge points is maintained. Fig. 9 shows the result after K-means clustering.

**Step 9: Computing and Validating**

The following equation is used to compute the position of the droplet:

\[ x_m = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad y_m = \frac{1}{n} \sum_{i=1}^{n} y_i \]  

\[ m_j = \frac{1}{N_j} \sum_{i \in w_j} x_j \]  

\[ D_j(x) = \| x - m_j \| \]   

\[ m_j = \frac{1}{N_j} \sum_{i \in w_j} x_j \]
The area surrounded by the detected edge points is the droplet. Since there are discontinuities in the detected edge points, to robustly estimate the droplet size, the following model is proposed:

\[ r = a_0 + a_1 \theta + a_2 \theta^2 \]  (11)

where \( r \) and \( \theta \) is the amplitude and angle of the vector which points to a point on the droplet edge from the computed center, and \( a_0 \), \( a_1 \) and \( a_2 \) are the coefficients which can be estimated using the Least Squares algorithm using the detected edge points \((x_i, y_i) \ (i = 1,..., N)\) or the corresponding \((r_i, \theta_i) \ (i = 1,..., N)\).

The size of the droplet can be measured by its area:

\[ A = \frac{1}{2} \int_{-\pi}^{\pi} r^2 d\theta = \frac{1}{2} \int_{-\pi}^{\pi} \left(a_0 + a_1 \theta + a_2 \theta^2\right)^2 d\theta \]  (12)

It should be pointed out that the proposed model for the droplet edge appears to be more appropriate than a circle or ellipse because it simply makes use of the fact that the actually droplet edge is smooth. In fact, the shape of the droplet is determined by a number of forces including the gravity, arc pressure and the surface tension. Because of the surface tension, the droplet maintains to be smooth but the effect of other forces make it impossible for the droplet to be an ideal circle or ellipse in the image. In addition, if an ellipse model is used, the coordinates of its center must be re-estimated in order to avoid large fitting error. However, as can be proved, the estimation of the center coordinates requires non-linear optimization and does not have an analytical solution. For model (11), the estimate can be obtained as an analytic solution and the center used to calculate \((r_i, \theta_i) \ (i = 1,..., N)\) does not affect the fitting accuracy. Hence, model (11) gives a robust yet computation effective method to estimate the size of the droplet.

In case a significant range of angle misses in the measurement data set \((r_i, \theta_i) \ (i = 1,..., N)\), a set of coefficients can still be estimated for model (11) to fit the given data set well. However, using the model to predict the edge in the missing angle range is to extrapolate and the accuracy can not be assured. Hence, to robustly estimate the size of the droplet which requires to compute in the whole angle range \([-\pi, \pi]\) as can be seen in Eq. 12, a model obtained using a data set with a large missing angle range should not be trusted and used. In this paper, \(\pi / 2 = 1.5708\) is taken as the maximally allowed missing angle range. If the missing angle range in the data set is larger than \(\pi / 2\), the model would fail the validation. Fig. 10 shows an example where the missing angle range is large and the fitted result. As can be seen, the model can fit the measurement data with high accuracy but its extrapolation appears not acceptable. Hence, the edge data set should be validated or the fitted model needs to be validated to assure that the estimate of the droplet size is robust. Because a droplet would appear in several images and its size should be calculated as an average, the proposed validation would not affect the availability of the computed size but improve the robustness of the estimation.

![Fig. 10 An example of the detected edge with a large missing angle range](image)

**Step 10: Transform the computed results into their original coordinates**

Since the size and the position are computed in the region of interest after bilinear interpolation, they need to be transformed back into their original coordinate system by using following equations:

\[ A_0 = A / 25 \]  (13)
\[ x_o = x_c - 10 + x_m / 5 \]  (14)
\[ y_o = y_c - 10 + y_m / 5 \]  (15)

**IV. RESULT AND DISCUSSION**

The authors have tested the proposed algorithm using a series of adjacent frames of metal transfer images. The used series of images are shown in Fig. 11. It contains 10 frames which describe the life span of a droplet. The first frame is the image when the droplet is just detached and the following frames record the travel of the detached droplet. In each frame, there are two droplets. The upper one is the new droplet which is just produced and being developed to detach from the wire tip. The lower one is the droplet which has been detached and is traveling.

![Fig.11 Ten adjacent frames](image)

To demonstrate the effectiveness of the proposed algorithm, a few intermediate results of the image processing are demonstrated. Figs. 12-15 show the image processing results for Frame 7 through frame 10. In all these figures, the original image,
bilinear-interpolated image, all detected edges, and edges after matching are shown. From these figures, it is seen that the detected edge of the droplet is sufficient to compute the position of the droplet with a high accuracy. Also the detected edge is enough to compute the size of the droplet. Figs. 16-19 plot the modeled edges for the data sets in Figs. 12-15.

![Figures 12-19 showing detected edges and fitted models](image)

Table 1 shows the actual positions and sizes of the lower droplet in these frames against their computations. Table 2 lists those for the upper droplet. The computed results which pass the validation are used to calculate the averages. As can be seen, the error for the lower droplet size is 0.44 pixel² or 0.6%. For the size of the upper droplet, the error is 3.5 pixel² or 7.7%. Those accuracies should be sufficient for the control of the droplet size. For the position, the accuracy appears also to be sufficient to estimate the trajectory of the flying droplet (lower droplet). Hence, the proposed algorithm appears has provided a robust estimate with sufficient accuracy for possible applications in droplet size and trajectory control.

Testing shows that the proposed algorithm needs about 2.5 seconds to process one frame in MatLab using a notebook with Intel® Pentium® M 1.86GHz processor and 1 GBytes RAM. Since the final real time processing system should be programmed in C, the processing time will be significantly reduced. Luo et al. [14] reported an over 100 times of speed improvement of compiled C over the interpreted MatLab. In [15], Tommiska et al. stated “This indicated complied (i.e., C source) code runs approximately 400 times faster than interpreted (i.e., MatLab) code” in their case. It is estimated that compiled C may process the image at 25 ms per frame. In the control system to be developed, the authors envision that a sequence of 30 images which contain a few metal transfer cycles be sampled in 10 ms at the frame rate of 3,000 frames per second and then be processed in approximately 750 ms. As a result, the control variable may be adjusted at a speed once per second which is acceptable for welding process control although further development of hardware and algorithm optimization would significantly improve the speed. Hence, the proposed algorithm appears to have the potential to be used in on-line control of metal transfer process.

V. CONCLUSIONS

- The bilinear interpolation is effective for image enhancement toward better edge detection.
- The proposed brightness based selection and edge based separation algorithm can detect droplets from the image and try to detect adequate edge information from interpolated images.
- The proposed model for droplet edge gives an effective method to estimate the size of the droplet robustly and accurately and the proposed model validation assures that the model used meets a minimal accuracy requirement.
- The speed of the image processing appears to meet the minimal requirement for real-time control.

It should be mentioned that certain algorithm parameters and equations are ad hoc for the well defined and constrained problem under investigation. In case welding conditions change, appropriate modifications may become necessary.

REFERENCES


### Table 1

<table>
<thead>
<tr>
<th>Frame Index</th>
<th>Droplet’ size</th>
<th>Droplet’ position</th>
<th>Computed size</th>
<th>Computed position</th>
<th>Size error</th>
<th>Position error</th>
<th>Max-Missing angle</th>
<th>Validity Passed?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame1</td>
<td>81</td>
<td>(135, 213)</td>
<td>93</td>
<td>(136, 213)</td>
<td>-12</td>
<td>(0, 0)</td>
<td>1.5644</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame2</td>
<td>68</td>
<td>(137, 213)</td>
<td>66</td>
<td>(135, 211)</td>
<td>2</td>
<td>(-2, -2)</td>
<td>0.9226</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame3</td>
<td>76</td>
<td>(138, 212)</td>
<td>76</td>
<td>(138, 211)</td>
<td>0</td>
<td>(0, -1)</td>
<td>0.6755</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame4</td>
<td>78</td>
<td>(140, 211)</td>
<td>76</td>
<td>(141, 210)</td>
<td>2</td>
<td>(1, -1)</td>
<td>0.6084</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame5</td>
<td>81</td>
<td>(142, 210)</td>
<td>97</td>
<td>(144, 209)</td>
<td>-16</td>
<td>(2, -1)</td>
<td>1.5895</td>
<td>No</td>
</tr>
<tr>
<td>Frame6</td>
<td>67</td>
<td>(144, 210)</td>
<td>57</td>
<td>(145, 209)</td>
<td>10</td>
<td>(1, -1)</td>
<td>0.8141</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame7</td>
<td>72</td>
<td>(145, 209)</td>
<td>72</td>
<td>(145, 209)</td>
<td>0</td>
<td>(0, 0)</td>
<td>0.5672</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame8</td>
<td>64</td>
<td>(146, 208)</td>
<td>64</td>
<td>(146, 208)</td>
<td>0</td>
<td>(0, 0)</td>
<td>0.5407</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame9</td>
<td>73</td>
<td>(148, 208)</td>
<td>72</td>
<td>(148, 208)</td>
<td>1</td>
<td>(0, 0)</td>
<td>0.5118</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame10</td>
<td>67</td>
<td>(149, 207)</td>
<td>66</td>
<td>(149, 207)</td>
<td>1</td>
<td>(0, 0)</td>
<td>0.7489</td>
<td>Yes</td>
</tr>
<tr>
<td>Average</td>
<td>71.7778</td>
<td></td>
<td>71.3333</td>
<td></td>
<td>0.44</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 2

<table>
<thead>
<tr>
<th>Frame Index</th>
<th>Droplet’ size</th>
<th>Droplet’ position</th>
<th>Computed size</th>
<th>Computed position</th>
<th>Size error</th>
<th>Position error</th>
<th>Max-Missing angle</th>
<th>Validity Passed?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame1</td>
<td>31</td>
<td>(132, 219)</td>
<td>26</td>
<td>(134, 217)</td>
<td>5</td>
<td>(2, -2)</td>
<td>1.1092</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame2</td>
<td>48</td>
<td>(133, 218)</td>
<td>52</td>
<td>(135, 218)</td>
<td>-4</td>
<td>(3, -3)</td>
<td>1.1715</td>
<td>No</td>
</tr>
<tr>
<td>Frame3</td>
<td>46</td>
<td>(133, 219)</td>
<td>35</td>
<td>(135, 216)</td>
<td>11</td>
<td>(3, -3)</td>
<td>1.8812</td>
<td>No</td>
</tr>
<tr>
<td>Frame4</td>
<td>47</td>
<td>(133, 219)</td>
<td>59</td>
<td>(135, 217)</td>
<td>-12</td>
<td>(2, -2)</td>
<td>1.0775</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame5</td>
<td>33</td>
<td>(134, 219)</td>
<td>34</td>
<td>(134, 219)</td>
<td>-1</td>
<td>(0, 0)</td>
<td>1.3189</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame6</td>
<td>48</td>
<td>(135, 218)</td>
<td>49</td>
<td>(131, 219)</td>
<td>-1</td>
<td>(-4, -1)</td>
<td>0.9813</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame7</td>
<td>34</td>
<td>(135, 218)</td>
<td>23</td>
<td>(135, 217)</td>
<td>11</td>
<td>(0, -1)</td>
<td>0.7556</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame8</td>
<td>49</td>
<td>(135, 218)</td>
<td>38</td>
<td>(135, 218)</td>
<td>11</td>
<td>(0, 0)</td>
<td>0.9450</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame9</td>
<td>51</td>
<td>(136, 218)</td>
<td>33</td>
<td>(137, 216)</td>
<td>18</td>
<td>(1, -2)</td>
<td>1.1508</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame10</td>
<td>70</td>
<td>(135, 218)</td>
<td>73</td>
<td>(134, 217)</td>
<td>-3</td>
<td>(-1, -1)</td>
<td>1.4146</td>
<td>Yes</td>
</tr>
<tr>
<td>Average</td>
<td>45.3750</td>
<td></td>
<td>41.8750</td>
<td></td>
<td>3.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>