Abstract

In this paper location M estimator is used to inspect the over and under fill liquid level of bottle in machine vision system. Different optimal edge detection algorithms such as Marr-Hildreth algorithm LoG, Canny algorithm and Shen Castan algorithm are used for liquid level inspection in industry until now. The filling level in a bottle is computed as an average distance from a specific reference line. The average operator is not robust to outliers in the data. In this paper, we propose to use the combination of location M estimator with any type of edge detection technique which will remove the need of a specific optimal edge detection technique and thus can result into easy hardware realization of the liquid level inspection algorithm.

Keywords: Quality control, Machine Vision, location M estimation, Optimal Edge detection, ISEF (Shen Castan algorithm), Canny Edge Detector, LoG.

1. Introduction

Machine vision system plays a vital role in manufacturing application, quality inspection and process monitoring as well. Traditionally, quality inspection is performed by trained human inspectors. In addition to being costly, this method is highly variable and decisions are not always consistent between inspectors or from day to day. This is, however, changing with the advent of electronic imaging systems and with the rapid decline in cost of computers, peripherals and other digital devices. Taking one application of inspection of bottle filling, the method is very fast, quiet repetitive and subjective in nature. In this type of environment, machine vision systems are ideally suited for routine inspection and quality assurance tasks. In machine vision based systems many edge detection techniques proposed by many researchers are prevailing. Each technique works nicely for the particular application only. There is not a general consensus about one or couple of methods to be used for edge detection in machine vision community. Significant work in bottle defect detection and in bottle filling level inspection cited literature is not available due to its inherently simple task of computing distance of the filling level with respect to a reference line. In [3] Y. Wang et al. proposed an algorithm for bottle finishing using Hough transform methods which would detect the defect bottle from the bulk and separate it out. In [4] Y. Wang et al. proposed watershed algorithm for bottle inspection by detecting out the possible defective regions in the upper portion of the bottle called the neck of the bottle and extracting these features from the image. Further for the purpose of classification the optimal hyper plane concept based on SVM method was used. In [5] Hui-MinMa et al. proposed an automatic inspection system based on eight CCD cameras which would give a decision about good or bad bottle based on top lead of the bottle. In [6] F.Daun et al. proposed a new algorithm stating that Hough transform and edge detection is a slow process so by analyzing the histogram of the edges of the bottle. Based on those edges an analysis was done on the shape and size of the bottle.

In [1] Kunal Pithadiya et al. proposed to use optimal edge detection techniques like ISEF [2] and Canny edge detection instead of template based edge detectors. The algorithm proposed in [1] computes average distance from a reference line. The average operation is susceptible to outliers and data variations. Motivated by the robust properties of M estimator [13], We propose to use location M estimator with any edge detection techniques like LoG, Canny edge detection and ISEF edge detection to inspect the filling level of the bottle. In section 2 we discuss the problem definition. In section 3 we discuss location M Estimation. In section 4 we propose different algorithms for filling level inspection using machine vision and results are also discussed in section 5. Section 6 concludes the paper.

2. Problem Definition

Filling bottle using machine with accuracy is subject to error from a wide variety of potential problems from flow rates to glass bottle variations. To ensure consistent fill levels 100% quality inspection is required. Inspection systems must also be capable of keeping up high speed filling/bottling machinery.
Figure 1: Outline of Bottle level filling system

[Courtesy: OMRON Tech. for focused Automation [6]]

Figure 1 shows a schematic of bottle filling system. Failure to properly fill bottles to the correct volumes as stated on packaging results in loss of customer loyalty, consumer fraud allegations and recalls. For instance if the milk bottles are not properly filled which are prescribed for the babies then the proper nutrition in the required amount would not reach to the baby’s body which results in loss of customer loyalty as well as fraud allegations. Overfilling results in giving away products and profits. The images of overfilled and under filled bottles are shown in Figure-2.

Figure 2: Over and under filled bottles

[Courtesy: OMRON Tech. for focused Automation [6]]

To compute the filling level Kunal Pithadiya et al. [1] proposed an algorithm based on region of interest (ROI) and average of the distances computed from distance of the filling level and a reference line. Most commonly used edge detection algorithms are template based edge detectors, LoG algorithm, Canny Edge detector and ISEF algorithm proposed by Shen and Castan. One of the template based edge detector is the Sobel edge detector, which uses templates in the form of convolution masks having the following values:

\[
\begin{align*}
&-1 & -2 & -1 \\
&S_y &= 0 & 0 & 0 & S_{xy} &= -2 & 0 & 2 \\
&1 & 2 & 1 & -1 & 0 & 1
\end{align*}
\]

The main disadvantage of these edge detectors is their dependence on the size of the object and they are having high sensitivity to noise, and Inaccurate too. The LoG algorithm is outline in Table I for which Locality is not especially good, and the edges are not always thin. Still, this edge detector is much better than the previous one in case of low signal to noise ratio. Malfunctioning at corners, curves and where the gray level intensity function varies, not finding the orientation of edge because of using the Laplacian filter.

<table>
<thead>
<tr>
<th>No.</th>
<th>Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Convolve the image I with a two dimensional Gaussian function.</td>
</tr>
<tr>
<td>2</td>
<td>Compute Laplacian of convolved image; call it L.</td>
</tr>
<tr>
<td>3</td>
<td>Edges pixels are those for which there is a zero crossing in L.</td>
</tr>
</tbody>
</table>

Two advanced and optimal edge detectors are Canny Edge Detector [8] and Shen and Castan’s Infinite Symmetric Exponential Filter (ISEF) [2]. Canny algorithm (Table 2) convolves the image with the derivative of a Gaussian.

<table>
<thead>
<tr>
<th>No.</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Read the image I.</td>
</tr>
<tr>
<td>2</td>
<td>Convolve a 1D Gaussian mask with I.</td>
</tr>
<tr>
<td>3</td>
<td>Create a 1D mask for the first derivative of the Gaussian in the x and y directions.</td>
</tr>
<tr>
<td>4</td>
<td>Convolve I with G along the rows to obtain Ix, and down the columns to obtain Iy.</td>
</tr>
<tr>
<td>5</td>
<td>Convolve Ix with Gx to have Ix’, and Iy with Gy to have Iy’.</td>
</tr>
<tr>
<td>6</td>
<td>Find the magnitude of the result at each pixel (x, y) using</td>
</tr>
</tbody>
</table>
\[
M(x, y) = \sqrt{(Ix(x,y))^2 + (Iy(x,y))^2}
\]

In ISEF algorithm (Table 3), the image is filtered by a recursive ISEF filter in X direction and in Y direction.

<table>
<thead>
<tr>
<th>No.</th>
<th>Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Apply ISEF Filter in X direction</td>
</tr>
<tr>
<td>2</td>
<td>Apply ISEF Filter in Y direction</td>
</tr>
<tr>
<td>3</td>
<td>Apply Binary Laplacian Technique</td>
</tr>
<tr>
<td>4</td>
<td>Apply Non Maxima Suppression</td>
</tr>
<tr>
<td>5</td>
<td>Find the Gradient</td>
</tr>
<tr>
<td>6</td>
<td>Apply Hysteresis Thresholding</td>
</tr>
<tr>
<td>7</td>
<td>Thinning</td>
</tr>
</tbody>
</table>

After applying the edge detection technique the distance between the reference line and the edge of filling level is computed as described [1]. The average operator is not robust to outliers. Consider an example of finding an average of a dataset having only one outlier. The effect of outlier on average operation can be seen in
figure 3, the average value is drifted towards the outlier data point. Due to the problem of outliers the distances computed by the algorithm given in [1] results in to different distance values for different edge detection techniques. Even the use of an optimal edge detection technique will not guarantee accurate results.

![Image of a filled bottle is captured using a CCD camera as shown in Fig-3. The image is cropped to make it a normalized image with respect to height of the conveyor belt. For the comparative study, different Edge](image)

**Figure 3 Effect of a single outlier on average operator**

3. Location M estimation

For a dataset $y$ having $n$ data points, the location model is given by $y = mx + c$ with $x$ equal to zero. To find the location estimate $\hat{y}$, the least squares method minimizes the summed square of residuals given by

$$E = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

(1)

The residual for the $i$th data point $e_i$ is defined as $e_i = y_i - \hat{y}_i$, where $\hat{y}_i$ is predicted value. The solution of equation (1) is

$$c = (\Sigma y_i)/n$$

(2)

The problem of average operator of equation (2) is that it is not robust towards outliers [12]. The estimate may tend to give erroneous results because its objective function $E = \sum_{i=1}^{n} e_i^2$ increases indefinitely with the residuals $e_i$. Hence, extreme outliers with arbitrarily large residuals can have an infinitely large influence on resulting estimate. Robust M-estimators attempt to limit the influence of outliers by replacing the square of the residuals with a less rapidly increasing loss function of the residuals $e_i$. Out of many loss functions, Tukey biweight objective function is especially resistant to observation on the extreme tails because the derivative function $\psi(.) = \rho(.)$ for Tukey function redescends to zero.

$$\rho_B(e) = \begin{cases} \frac{2}{k} \left[ 1 - \left( \frac{1 - e^2}{k} \right)^{\frac{1}{2}} \right]^3 & \text{for } |e| \leq k \\ \frac{2}{k} \left[ 1 - \left( \frac{1 - e^2}{k} \right)^{\frac{1}{2}} \right] & \text{for } |e| > k \end{cases}$$

(3)

The tuning constant $k$, is expressed as a multiple of scale of $y_i$, $k=CS$, where $S$ is the measure of the scale of $y$. Since the standard deviation is more influenced by extreme observations we use the median absolute deviation (MAD), $S = \text{Median} |y_i - \hat{y}_i|$. To further normalize the median with respect to population we use $S = \text{MAD}/0.6745$. For Tukey biweight function $k=4.685*S$ produces 95% efficiency. The general M-estimator minimizes the following objective function

$$\sum_{i=1}^{n} \rho_B(e_i) = \sum_{i=1}^{n} \rho_B(y_i - \hat{y}_i), \hat{y}_i = c + e_i$$

(4)

Differentiating the objective function of equation (4) with respect to $c$ and setting the partial derivatives equal to 0, will result into system of equation given by $\psi(.) = \rho(.)$. Defining the weight function $w(e) = \psi(e)/e$ and solving the estimating equations is a weighted least-squares problem, minimizing $\sum w_i e_i^2$.

The weights depend upon the residuals and residuals depend upon estimated coefficients which again depend upon the weights. An iterative algorithm called as iteratively reweighed least squares, IRLS is required, which is outlined below [3].

1. Select initial estimates $C^{(0)}$, such as the least squares estimates.
2. At each iteration $t$, compute $e_i^{(t-1)}$ and weights $w_i^{(t-1)} = w \left[ e_i^{(t-1)} \right]$ from previous iteration.
3. Solve for new weighted-least-squares estimate $e^{(t)}$. Steps 2. and 3. are repeated until the estimated coefficients converge.

4. Proposed Algorithm

Image of a filled bottle is captured using a CCD camera as shown in Fig-3. The image is cropped to make it a normalized image with respect to height of the conveyor belt. For the comparative study, different Edge
detection techniques like Canny, LoG and ISEF Edge detection techniques can be applied. The required steps for each edge detection algorithms are given in Table 1, 2 and 3 respectively. We propose an algorithm to decide the over/under filling level of bottle based on location M estimation which is outlined in Table 4.

Table 4: Proposed algorithm to find distance using robust location M estimation

<table>
<thead>
<tr>
<th>No.</th>
<th>Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Capture the image.</td>
</tr>
<tr>
<td>2</td>
<td>Apply an optimal Edge detection Algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Find a region of interest (ROI)</td>
</tr>
<tr>
<td>4</td>
<td>Decide a horizontal ROI as shown in Figure 5.</td>
</tr>
<tr>
<td>5</td>
<td>Take the bottle neck bottom edge as a reference line as shown in Figure 6.</td>
</tr>
<tr>
<td>6</td>
<td>For every pixel of value 1, in box 1 as shown in Figure 6, find a pixel in box 2 having value 1.</td>
</tr>
<tr>
<td>7</td>
<td>Join the two pixels by vertical line i.e. find the distance between these two pixels. Do it for all the pixels having value 1, in both boxes 1 and 2. Figure 7 contains this type of vertical lines.</td>
</tr>
<tr>
<td>8</td>
<td>Apply robust M location estimate to the distance dataset to find the estimate of distance.</td>
</tr>
<tr>
<td>9</td>
<td>If the estimated distance is greater than the datum line (shown in Figure 8 with red colour), then the bottle is over filled but if it is less than the datum line, then the bottle is under filled.</td>
</tr>
</tbody>
</table>

Figure 5: Image with Region of Interest
Figure 6: Image with Regions in BOX 1 & BOX 2
Figure 7: Image with distance lines
5. Results

After applying the proposed algorithm we get the distances as given in appendix. The avg. distances are available as shown in Table-5 for each edge detection algorithm. The variations of distances which we got after applying the proposed algorithm for each edge detection technique are shown in Figure-12, 13 & 14. All three type of optimal edge detector provides the information about the level of the bottle approximately with one mm accuracy. We have proposed the algorithm of robust Location M estimation with any one optimal edge detector from the above list is proved in the Table 5 with having almost same avg. distance for all three. The proposed algorithm gives you freedom to choose any one type of edge detector from the above list with robust location M estimation technique for the optimal solution of bottle filling level inspection.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Reference</th>
<th>LOG</th>
<th>Canny</th>
<th>ISEF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Average operator</td>
<td>In pixel</td>
<td>101</td>
<td>106</td>
<td>107</td>
</tr>
<tr>
<td></td>
<td>In mm</td>
<td>35.68</td>
<td>37.30</td>
<td>37.63</td>
</tr>
<tr>
<td>Robust M location estimation</td>
<td>In pixel</td>
<td>109</td>
<td>109</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>In mm</td>
<td>38.29</td>
<td>38.61</td>
<td>38.69</td>
</tr>
</tbody>
</table>

6. Conclusion

In the result we have compared the performance of Location M estimation with different edge detection techniques for a simple application of water bottle level
filling inspection using machine vision system. Location M estimation with ISEF edge detection technique is optimal the solution for distance estimation. From the result it is been concluded that, if the robust location M estimation is used with any type of edge detector then the obtained results are very good and can give better decision about the over and under filled bottle.

7. References:


[6] OMRON Tech. focused for automation solutions


### APPENDIX

<table>
<thead>
<tr>
<th>Distance</th>
<th>Canny</th>
<th>Log</th>
<th>ISEF Distance</th>
<th>Canny</th>
<th>Log</th>
<th>ISEF</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>55</td>
<td>54</td>
<td>55</td>
<td>d79</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d2</td>
<td>55</td>
<td>54</td>
<td>55</td>
<td>d80</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d3</td>
<td>55</td>
<td>54</td>
<td>55</td>
<td>d81</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d4</td>
<td>55</td>
<td>54</td>
<td>55</td>
<td>d82</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d5</td>
<td>55</td>
<td>54</td>
<td>55</td>
<td>d83</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d6</td>
<td>55</td>
<td>71</td>
<td>55</td>
<td>d84</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d7</td>
<td>55</td>
<td>70</td>
<td>55</td>
<td>d85</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d8</td>
<td>55</td>
<td>69</td>
<td>55</td>
<td>d86</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d9</td>
<td>111</td>
<td>69</td>
<td>113</td>
<td>d87</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d10</td>
<td>107</td>
<td>105</td>
<td>108</td>
<td>d88</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d11</td>
<td>106</td>
<td>104</td>
<td>107</td>
<td>d89</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d12</td>
<td>105</td>
<td>103</td>
<td>106</td>
<td>d90</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d13</td>
<td>105</td>
<td>103</td>
<td>105</td>
<td>d91</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d14</td>
<td>104</td>
<td>103</td>
<td>106</td>
<td>d92</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d15</td>
<td>104</td>
<td>60</td>
<td>105</td>
<td>d93</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d16</td>
<td>104</td>
<td>91</td>
<td>104</td>
<td>d94</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d17</td>
<td>103</td>
<td>97</td>
<td>103</td>
<td>d95</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d18</td>
<td>103</td>
<td>98</td>
<td>103</td>
<td>d96</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d19</td>
<td>103</td>
<td>113</td>
<td>103</td>
<td>d97</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d20</td>
<td>104</td>
<td>112</td>
<td>104</td>
<td>d98</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d21</td>
<td>55</td>
<td>69</td>
<td>106</td>
<td>d99</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d22</td>
<td>107</td>
<td>92</td>
<td>106</td>
<td>d100</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d23</td>
<td>107</td>
<td>106</td>
<td>102</td>
<td>d101</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d24</td>
<td>107</td>
<td>54</td>
<td>103</td>
<td>d102</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d25</td>
<td>106</td>
<td>68</td>
<td>104</td>
<td>d103</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d26</td>
<td>106</td>
<td>95</td>
<td>106</td>
<td>d104</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d27</td>
<td>106</td>
<td>76</td>
<td>107</td>
<td>d105</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d28</td>
<td>107</td>
<td>75</td>
<td>107</td>
<td>d106</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d29</td>
<td>54</td>
<td>75</td>
<td>107</td>
<td>d107</td>
<td>110</td>
<td>112</td>
</tr>
<tr>
<td>d30</td>
<td>109</td>
<td>109</td>
<td>108</td>
<td>d108</td>
<td>110</td>
<td>112</td>
</tr>
<tr>
<td>d31</td>
<td>110</td>
<td>72</td>
<td>108</td>
<td>d109</td>
<td>110</td>
<td>112</td>
</tr>
<tr>
<td>d32</td>
<td>108</td>
<td>69</td>
<td>103</td>
<td>d110</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d33</td>
<td>107</td>
<td>67</td>
<td>107</td>
<td>d111</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d34</td>
<td>107</td>
<td>94</td>
<td>107</td>
<td>d112</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d35</td>
<td>107</td>
<td>54</td>
<td>107</td>
<td>d113</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d36</td>
<td>107</td>
<td>54</td>
<td>107</td>
<td>d114</td>
<td>110</td>
<td>109</td>
</tr>
<tr>
<td>d37</td>
<td>108</td>
<td>106</td>
<td>107</td>
<td>d115</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d38</td>
<td>108</td>
<td>107</td>
<td>108</td>
<td>d116</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d39</td>
<td>109</td>
<td>107</td>
<td>109</td>
<td>d117</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d40</td>
<td>108</td>
<td>108</td>
<td>108</td>
<td>d118</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d41</td>
<td>108</td>
<td>107</td>
<td>108</td>
<td>d119</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d42</td>
<td>108</td>
<td>107</td>
<td>108</td>
<td>d120</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d43</td>
<td>108</td>
<td>107</td>
<td>108</td>
<td>d130</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d44</td>
<td>108</td>
<td>107</td>
<td>109</td>
<td>d131</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d45</td>
<td>108</td>
<td>107</td>
<td>109</td>
<td>d132</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d46</td>
<td>108</td>
<td>107</td>
<td>110</td>
<td>d133</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d47</td>
<td>108</td>
<td>108</td>
<td>109</td>
<td>d134</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d48</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d135</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d49</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d136</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d50</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d137</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d51</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d138</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d52</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d139</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d53</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d140</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d54</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d141</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d55</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d142</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d56</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d143</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d57</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d144</td>
<td>111</td>
<td>109</td>
</tr>
<tr>
<td>d58</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d145</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d59</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d151</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>d60</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d152</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d61</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d153</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d62</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d154</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d63</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d155</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d64</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d156</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d65</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d157</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d66</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d158</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d67</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d159</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d68</td>
<td>110</td>
<td>108</td>
<td>109</td>
<td>d160</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>d69</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d161</td>
<td>110</td>
<td>88</td>
</tr>
<tr>
<td>d70</td>
<td>109</td>
<td>108</td>
<td>109</td>
<td>d162</td>
<td>55</td>
<td>99</td>
</tr>
</tbody>
</table>