Paying Attention to Symmetrical Interest Points
Abstract
Most visual Simultaneous Localization And Mapping (SLAM) methods use interest points as landmarks in their maps of the environment. Often the interest points are detected using contrast features, for instance those of the Scale-Invariant Feature Transform (SIFT). The SIFT interest points, however, have problems with stability and noise robustness. Taking inspiration from human vision, the use of local symmetry to select interest points is proposed. Symmetry is a stimulus that occurs frequently in everyday environments where our robots operate in, making it useful for SLAM. Furthermore, symmetrical forms are inherently redundant, and can therefore be more robustly detected. The proposed method, the MUlti-scale Symmetry Transform (MUST), has been tested on stability, robustness, and repeatability. Moreover, the method is used to select landmarks to represent the environment. To test the SLAM performance of our model, we recorded a SLAM database with a mobile robot, and annotated the database by manually adding ground-truth positions. The results show that interest points selected using symmetry are more stable and more robust to noise and contrast manipulations, have a slightly better repeatability, and above all, result in better overall SLAM performance.

This chapter is based on:

8.1 Introduction

One of the fundamental tasks of an autonomous robot is to build a map of the environment and use it for self-localization. The problem of Simultaneous Localization and Mapping (SLAM) has therefore received much attention in the last decade (Thrun et al., 2005). Nowadays, approaches using laser range finders are very successful. SLAM using vision, however, remains a challenging research topic, (e.g., Frintrop & Jensfelt, 2008; Davison et al., 2007).

Using a camera has the advantage over a laser-range finder that it is a passive sensor that is low cost, low power, and lightweight. A camera furthermore provides a rich source of information, which enables the use of sophisticated detection and recognition methods. The difficulty, however, is to extract relevant information from the high-dimensional visual data in real time. In this chapter, the use of local symmetry to select relevant visual information is proposed.

Most visual SLAM systems use visual landmarks to create a map of the robot’s environment. It is important to select robust, stable landmarks that will be recognizable in future encounters. Furthermore, the number of selected landmarks should be limited, since the computational complexity of the SLAM algorithms strongly depends on the number of landmarks.

A common approach for the selection of landmarks in current visual-SLAM systems is to detect interest point in the camera images (Mozos et al., 2008). Most approaches use contrast features to detect interest points. Examples are the Scale-Invariant Feature Transform (SIFT) (Lowe, 2004; Se et al., 2002), Speeded-Up Robust Features (SURF) (Bay et al., 2006; Murillo et al., 2007), and Harris corners (Davison & Murray, 2002). In this chapter, the use of local symmetry to detect interest points is suggested. The proposed MUlti-scale Symmetry Transform (MUST) is compared with SIFT, since it is among the best performing interest point detectors in SLAM (Mozos et al., 2008), as well as in object recognition (Moreels & Perona, 2007).

Although systems using SIFT have successfully been applied to SLAM, there are three important drawbacks. The interest points are very susceptible to noise, not all selected landmarks are recognized when the robot returns to a previously visited location, and too many interest points are found in an image, but only few points are so stable that they can be tracked over a number of successive frames. Thus SIFT has problems with robustness, repeatability, and stability.
In this chapter, a landmark-selection mechanism is proposed that uses local symmetries instead of local contrast. As argued in Section 6.4, symmetry methods have fewer problems with noisy conditions than contrast methods. The use of local symmetry is therefore hypothesized to perform better regarding the problems mentioned above. Our choice for symmetry is motivated by human behavior. As has been discussed in Chapter 2, symmetry is detected very rapidly, especially when patterns have multiple axes of symmetry (Palmer & Hemenway, 1978). Moreover, humans pay attention to locally symmetric parts of images (see Chapter 3). Furthermore, humans use symmetry to segregate a figure from its background (Driver et al., 1992). These findings suggest that symmetry is used in preattentive vision, and can therefore be used for context-free object segmentation and landmark selection. Assuming that the human visual system has evolved to be as effective as possible in the kinds of environments that humans have to operate in, this suggests that symmetry detection might be useful in robots that have to operate in similar environments.

Using symmetry to select landmarks exploits the fact that most man-made indoor environments contain many symmetrical objects and forms. Since symmetry is a strong non-accidental property, we believe that its use will result in the selection of valuable landmarks for the visual SLAM system.

To detect local symmetry in an image, a number of symmetry operators exist. Reisfeld et al. (1995), for instance, developed a mirror-symmetry operator by comparing the gradients of neighboring pixels. Heidemann (2004) extended this work to the color domain. Reisfeld et al. also proposed a radial-symmetry operator that promotes patterns that are symmetric in multiple symmetry axes. A faster operator to detect radial symmetry, the Fast Radial Symmetry Transform (FRST), is proposed in (Loy & Zelinsky, 2003).

In this chapter, a novel scale- and rotation-invariant interest-point detector is proposed, which is called the MUlti-scale Symmetry Transform (MUST). The detector extends the FRST symmetry operator to a model that detects interest points on multiple scales. Techniques from SIFT are used to obtain a scale- and rotation-invariant representation of the interest points. The use of MUST for selecting landmarks for visual SLAM is discussed. The results show that landmarks selected using local symmetry are more robust to noise and have a higher repeatability, and require fewer computations. Most importantly, the overall performance of the SLAM system increases when interest points are selected using local symmetry instead of SIFT.
8.2 Methods

The complete system consists of three parts. The first part is the selection of interest points based on local symmetry. The detected interest points are then fed to a visual buffer to select stable interest points as landmarks. The selected landmarks are finally used by the EKF-SLAM system to build a map of the environment and to estimate the position of the robot in the environment.

8.2.1 Interest Points Based on Local Symmetry

As a basis of our interest point detector, we use the Fast Radial Symmetry Transform (Loy & Zelinsky, 2003), and extended it to a multi-scale and rotation-invariant detector, MUST. The basis of the FRST is given in Figure 8.1, and MUST is depicted in Figure 8.2.

To obtain a multi-scale interest-point detector, we use a pyramid approach similar to that used in (Lowe, 2004). The symmetry response is calculated at five spatial octaves,
\( \emptyset = \{-1, 0, 1, 2, 3\} \). In the first octave, -1, the gray-scaled input image, \( I_{-1} \), has twice the resolution of the original image, similar to (Lowe, 2004). For each next octave, the input image of the previous octave is smoothed with a Gaussian kernel and down-sampled by a factor of two. Within each octave, there are three scales, \( s \in \{0, 1, 2\} \), with progressive smoothing. This gives a pyramid of gray-scaled images, \( I_{o,s} \), for a given octave \( o \) and scale \( s \).

The symmetry transform, \( \Phi(o,s) \), for octave \( o \) and scale \( s \), is calculated by investigating the local radial symmetry at a set of different radii. The size of the radii depends on the scale \( s \). The set of radii used to calculate the symmetry response is defined as \( R_s = (1 + s/(s_{\text{max}} - 1)) \cdot R \), where \( s_{\text{max}} = 3 \) and \( R = \{1, 3, 5\} \). The symmetry transform is then:

\[
\Phi(o,s) = \frac{1}{|R_s|} \sum_{r \in R_s} \psi_r(o,s)
\]

where \( \psi_r(o,s) \) is the symmetry response at octave \( o \) and scale \( s \) for radius \( r \).

\( \psi_r(o,s) \) is determined by first calculating the gradients of the input image \( I_{o,s} \) with horizontally and vertically aligned Sobel filters, resulting in a magnitude map, \( G_{o,s} \), and an orientation map, \( \Theta_{o,s} \). Then, each pixel \( p \) votes for the existence of symmetry based on its gradient, \( \Theta_{o,s}(p) \), its magnitude, \( G_{o,s}(p) \) and the given radius \( r \). This is related to the Hough transform. A pixel votes for the existence of both a bright symmetrical form on a dark background, and a dark symmetrical form on a bright background at respectively location \( p_+ \) and \( p_- \):

\[
p_+ = p + [r \cdot (\cos \Theta_{o,s}(p), \sin \Theta_{o,s}(p))] \quad (8.2)
\]
\[
p_- = p - [r \cdot (\cos \Theta_{o,s}(p), \sin \Theta_{o,s}(p))] \quad (8.3)
\]

where \([\ldots]\) is the nearest-integer function. Subsequently, an orientation projection map, \( O_r \), is calculated that counts the number of symmetry votes, as well as a magnitude projection map, \( M_r \), which keeps track of the magnitudes of the gradients that contribute to these votes. Initially, all values in these maps are set to zero. Then, for every pixel \( p \), the maps are updated according to:

\[
O_r(p_+) = O_r(p_+) + 1, \quad O_r(p_-) = O_r(p_-) - 1 \quad (8.4)
\]
\[
M_r(p_+) = M_r(p_+) + G_{o,s}(p), \quad M_r(p_-) = M_r(p_-) - G_{o,s}(p) \quad (8.5)
\]
8.2. Methods

**Figure 8.2:** MUST, the MUlti-scale Symmetry Transform. Symmetry is calculated at multiple octaves and scales. At the first octave, $o = -1$, the image is double its original size. For every next octave, the image is down-scaled by a factor two. A total of five octaves is used. Within an octave there are three scales with progressive Gaussian blurring. For every image in the image pyramid, the symmetry response, $\psi_r(o,s)$, is calculated at three different radii by applying the symmetry operator. The average of these responses results in the symmetry map, $\Psi(o,s)$, for the given octave $o$ and scale $s$. In the last step, the interest points are obtained by finding points that have a local maximal or minimal symmetry value. The local neighborhood of such an interest point is described by the SIFT descriptor. The size of the neighborhood depends on the scale.

Finally, the symmetry map for radius $r$ at octave $o$ and scale $s$ is determined by:

$$\psi_r(o,s) = F_r \ast A_r$$  \hspace{1cm} (8.6)

and

$$F_r(p) = M_r(p) \cdot O_r(p)/k_r$$  \hspace{1cm} (8.7)

where $O_r(p)$ has a upper value of $k_r$, and a lower value of $-k_r$. $k_r$ has been experimentally established in (Loy & Zelinsky, 2003) at $k_r = 8$ for $r = 1$, and $k_r = 9.9$ otherwise. $A_r$ is a Gaussian kernel of size $r \times r$ with a standard deviation of $0.25r$.

Equation (8.7) weighs the votes in the orientation projection map with the values in
the magnitude projection map. This results in stronger symmetry votes for stronger
gradients. The convolution with the Gaussian kernel in Equation (8.6) spreads the
symmetry votes over neighboring pixels, with a larger spread for larger radii. This
allows symmetrical patterns to deviations from perfect radial symmetry.

The above gives us the symmetry response $\psi_r(o,s)$ for the radius $r$ at octave $o$ and scale
$s$. By averaging the symmetry responses over all radii in $R_s$, according to equation
(8.1), we obtain the full symmetry response at octave $o$ and scale $s$, $\Psi(o,s)$.

Next, the interest points in every octave and scale are determined by finding the points
in $\Psi(o,s)$ that are either a maximum or a minimum in the spatial neighborhood of
$11 \times 11$ pixels. Different from SIFT, the points do not need to have an optimal value
over neighboring scales, since this resulted in the rejection of too many valuable interest
points. A pixel can therefore potentially hold multiple interest points at different scales,
which is fine because each corresponds to a different symmetrical pattern.

For each interest point $i$ found in the symmetry maps the following information is
stored:

- the location of the interest point in the original resolution of the input image,
  $x_i = (x_i, y_i)$,
- the scale value, $\sigma_i = 2^{o_i+s_i/s_{\text{max}}}$,
- the symmetry strength, $v_i = \Psi(o_i,s_i,x_i)$,
- the orientation, $\gamma_i$, and
- the descriptor of the interest point, $d_i$.

The later two are described in the next paragraph.

To calculate the orientation and the descriptor of the interest points, we use the corre-
sponding methods of the SIFT algorithm (Lowe, 2004). The orientation of the interest
point is determined by finding the dominant gradient in the local neighborhood of the
interest point. This orientation is used to obtain a rotationally invariant descriptor of
the local neighborhood patch. The neighborhood is described by histograms of gra-
dients. The size of the neighborhood depends on the scale value of the interest point.
This makes that the descriptor is also scale invariant. To calculate the histograms of
gradients, the patch is divided in $4 \times 4$ squares. A histogram of gradients is then
calculated for each square. Since there are 16 squares, and each histogram contains 8
bins, this gives us a feature vector with 128 values. The magnitude of the feature vector
8.2. Methods

Figure 8.3: Two examples images with the MUST and SIFT interest points. The black circles present the stable interest points that are found in both the current and the previous image. The white circles depict the unstable interest points that are only found in the current image. The proportion of stable points is given at the bottom.

is normalized to 512. For more detailed information about the method to calculate the descriptor, we refer to (Lowe, 2004).

Figure 8.3 shows the interest points found by MUST on two of the images used in our experiments as compared to the SIFT interest points. Note that SIFT results in a large number of interest points, of which many are unstable. MUST, on the other hand, results in less, but more stable points.

To summarize, MUST calculates symmetry maps on multiple octaves and scales of the image using the symmetry transform. In these symmetry maps, we find points that have a locally optimal (i.e., maximal or minimal) symmetry value. These points are the interest points our method returns. We then use the SIFT descriptor to describe the interest points. We thus replace the SIFT method to find interest points based on difference of Gaussians by MUST, that finds interest points based on local symmetry.

8.2.2 The Visual Buffer

Both MUST and SIFT result in a large number of interest points when applied to the camera images taken by the robot. In our experiments, MUST detects on average 40 interest points per image, and SIFT 124. Using all these points would result in far too many landmarks in the complete map. To be practically usable, the state matrix of the Extended Kalman Filter should maximally contain a few hundred landmarks in total. Furthermore, most interest points found in one observation are not detected in subsequent observations. In other words, many interest points are unstable or only detectable...
from a particular viewpoint, and are therefore useless for SLAM. We propose to use a
visual buffer to retain only stable interest points, similar to (Frintrop & Jensfelt, 2008;
Se et al., 2002).

The buffer contains the last $N$ camera images. The interest points in the current image
are compared to those in the $N - 1$ previous images. An interest point $i$ in the current
observation is selected as landmark if it satisfies the following criteria:

1. the interest point is matched in $K$ of the previous images in the buffer, and $K \geq M$,
   where $M$ is the minimally necessary number of matching images,
2. the estimates of the position of the landmark in the environment are congruent,
   and
3. to avoid spurious interest points, the strength of the interest point, $v_i$, is at least
   a proportion of the strength of the maximum interest point that is successfully
   matched in the current image:

   $$v_i \geq \lambda \cdot v_{\text{max}}$$  \hspace{1cm} (8.8)

For criterion 1, the interest point with descriptor $d_i$ is matched when there is an interest
point in the previous image with a descriptor $d_j$ that is sufficiently similar. This is true
when the Euclidean distance is below the threshold $\tau_1$:

$$||d_i - d_j|| < \tau_1$$  \hspace{1cm} (8.9)

Additionally, to ensure unique interest points, the best-to-next-best ratio should be
smaller then the threshold $\delta_1$:

$$||d_i - d_j||/||d_i - d_l|| < \delta_1$$  \hspace{1cm} (8.10)

where $d_l$ is the descriptor of the second most similar interest point in the previous
image. This prevents ambiguous landmarks in the database.

For criterion 2, the landmark’s position is estimated by triangulation using the bearings
of the interest point in the current image and the previous images, and the displacement
of the robot. This results in a set of estimates of the range and bearing,

$$P = \{p_k | p_k = \langle r_k, \theta_k \rangle \land 1 \leq k \leq K\}$$  \hspace{1cm} (8.11)
The landmark is accepted if
\[
\text{var}(R) < \rho
\]  
(8.12)
where \(\text{var}\) is the variance, and \(R = r_k\).

The visual buffer tests the quality of the interest points, and adds only strong and stable points that are observable from multiple viewpoints to the map. An additional benefit is that the covariance matrix of the observation error, used in the EKF, can be initialized based upon the covariance matrix of the estimated landmark positions, \(\text{cov}(P)\).

### 8.2.3 Visual SLAM

We used a standard implementation of the Extended Kalman Filter (EKF) as basis of the SLAM system (Durrant-Whyte & Bailey, 2006), as discussed in Section 6.5.4. A full description of EKF-SLAM is given in Appendix C. This section focuses on the incorporation of the landmark observations in EKF-SLAM.

A landmark \(i\) that results from the buffer is classified as either a new landmark, or a previously observed landmark that is already in the map. It concerns a previously observed landmark if the landmark in the database with the most similar descriptor, \(j\), fulfills three criteria:

1. the distance between the descriptors of landmark \(i\) and landmark \(j\) is smaller then a given threshold \(\tau_2\):
\[
||d_i - d_j|| < \tau_2
\]  
(8.13)
2. the best-to-next-best ratio is smaller then a given threshold \(\delta_2\):
\[
||d_i - d_j||/||d_i - d_l|| < \delta_2
\]  
(8.14)

where \(d_i\) is the descriptor of the currently observed landmark, \(d_j\) is the most similar descriptor, and \(d_l\) is the second most similar descriptor in the database.

3. the position of the current observation should be within the vicinity of the landmark in the database:
\[
\sqrt{(x_i - x_j)^T S_j^{-1}(x_i - x_j)} < \eta
\]  
(8.15)
where $S_j$ is the uncertainty covariance matrix, which is discussed in the next paragraph. The landmark is classified as new only if none of the three criteria is fulfilled. This ensures uniqueness of a new landmark and thus avoids ambiguous landmarks in the map.

The landmark is classified as new only if none of the three criteria is fulfilled. For a new landmark, the state matrix and the state covariance matrix are augmented using the observation, $z_i$, and the uncertainty covariance matrix, $S_i$, where $z_i$ is set to mean(P). $S_i$ is determined using cov(P) and the uncertainty of the robot’s position in the EKF. Additionally, the descriptor of the interest point is stored. A landmark that is matched with an existing landmark in the database is used in the update step of the EKF.

Because EKF is quadratic in the number of landmarks, we restrict the size of the map to a maximum of 350 landmarks, so the system can run in real time. More efficient implementations exist that can handle larger numbers of landmarks (Guivant & Nebot, 2001). This, however, lies outside of the scope of this research. Important is that the benefits of using symmetry to detect landmarks also hold for other implementations of the EKF, as well as for Particle Filters (e.g., Sim et al., 2007; Montemerlo et al., 2003) and the Information Filter (Thrun et al., 2004).

### 8.3 Experiments

We performed a number of experiments with a Pioneer II DX robot equipped with a Sony D31 camera to test the use of local symmetry for visual SLAM and to compare it to using standard SIFT. To be able to repeat the experiments, we created a SLAM database. This database contains camera images and odometry information, along with the ground truth position of the robot. The data was recorded in ten different runs, in which the robot drove four laps of approximately 35 meters through an office environment and hallway with an average speed of 0.3 m/s. Camera images of 320 x 240 pixels were stored at 5 Hz. At intervals of one meter, the true location of the robot was logged by hand. This enabled us to quantify the performance of the SLAM estimation.

SIFT and MUST result in different types of interest points, which might have an influence on the best settings for the other parameters in the SLAM system. We therefore optimized the parameters for the visual buffer ($N, M, \tau_1, \delta_1, \text{and } \rho$) and for the SLAM
system ($\tau_2$ and $\delta_2$) for both MUST and SIFT separately. Any differences in performance can therefore be subscribed to the interest-point detectors.

### 8.3.1 Stability

The stability of the interest points has been tested on 200 images in multiple sequences recorded by the robot. For every image, the stability of all interest points is tested. An interest point is considered stable if it is found in all previous $N$ consecutive images. If the interest point is not found in all of the previous images, it is considered unstable. It must be noted that the visual buffer is not used in this experiment but all interest points detected by the MUST and SIFT are used.

### 8.3.2 Robustness

Cameras that robots use to perceive the world are usually noisy causing different kinds of perturbations of the camera images. Furthermore, most environments that a robot needs to map are subject to changing light conditions. It is therefore important that a landmark selection mechanism is robust to these confounding factors.

In this experiment, 57 camera images were used from one of the runs in the database. The images were taken at intervals of approximately 3 meters. To test the robustness to noise, we added Gaussian pixel noise, and Gaussian smoothing to the original images. In addition, we manipulated the contrast and brightness to test the robustness to changing light conditions. The visual buffer is not used and all interest points detected by MUST and SIFT are used. The functions used for the manipulations are:

1. **Gaussian pixel noise**: the original image is transformed to:

   \[
   I'(x,y) = I(x,y) + X(\alpha_g) \tag{8.16}
   \]

   where $I(x,y) \in [0,1]$ is the intensity of pixel $(x,y)$, and $X(\alpha_g)$ is a random sample from the normal distribution $N(0, \alpha_g^2)$.

2. b) **Gaussian smoothing**: the original image is convolved with a Gaussian mask:

   \[
   I' = I \ast G_s \tag{8.17}
   \]
where $G_s$ is a Gaussian mask of size $s \times s$, with a standard deviation of $\sigma = s/6$.

3. **Contrast manipulation:**

$$I'(x,y) = I(x,y) + \alpha_c (I(x,y) - \bar{I}_{x,y})$$  \hspace{1cm} (8.18)

where $\bar{I}_{x,y}$ is the local average in a neighborhood of $21 \times 21$ pixels around pixel $(x,y)$. The contrast increases with positive values for $\alpha_c$, and decreases for negative values.

4. **Brightness manipulation:**

$$I'(x,y) = I(x,y)^{\log \alpha_b/\log 0.5}$$  \hspace{1cm} (8.19)

For $\alpha > 0.5$, the pixels are brightened, for $\alpha < 0.5$, the pixels are darkened.

The reason to not use the dataset in (Mikolajczyk, Tuytelaars, Schmid, Zisserman, Matas, Schaffalitzky, Kadir, & Van Gool, 2005) is that it contains images of outdoor scenes, which contain fewer symmetrical objects than indoor environments.

The robustness is measured by the proportion of matching interest points between the original and the manipulated images. Two interest points match when criterion 1 of the visual buffer is met (see section 8.2.2), where $\tau_1 = 0.6$ and $\delta_1 = 0.75$. Additionally, the spatial distance between the two interest points in the image should be fewer than 3 pixels.

### 8.3.3 Repeatability

For good SLAM performance, it is crucial that landmarks added to the map are observable on future encounters. We therefore test the repeatability of the interest points. For every experimental run, we selected three parts of the robot’s trajectory, and aligned the sequences of images of the four consecutive laps that are approximately at the same position. The interest points in the images in the first lap are then compared to the images taken at approximately the same position in the later laps.

The average proportion of interest points that are matched in lap 2, 3, and 4 is the measure of repeatability. An interest point matches when criterion 1 of the visual buffer is met (see section 8.2.2), where $\tau_1 = 0.6$ and $\delta_1 = 0.75$. Unlike the evaluation
8.4. Results

Figure 8.4: Stability of the interest points as a function of the number of consecutive images. The graphs show the proportions of interest points that are stably found in all consecutive images.

of the robustness, we do not put a constraint on the spatial distance between the two observed interest points, since there is variation in the exact position and orientation of the robot, which may cause relatively large displacements of the interest points.

8.3.4 Visual SLAM Performance

Because we logged the ground-truth position of the robot at intervals of 1 meter, we can quantify the performance of the complete visual SLAM system. For that, we compare the estimated position of the robot made by the EKF with the ground-truth position. The SLAM performance is the average Euclidean distance between the estimated and ground-truth positions in the last of the four laps.

8.4 Results

8.4.1 Stability

Figure 8.4 shows the proportion of stable interest points as a function of the number of consecutive images. It can be seen that the symmetrical interest points selected by
Chapter 8. Paying Attention to Symmetrical Interest Points

Figure 8.5: Robustness to noise and changing light condition. The lines show the mean proportion of matching interest points between the distorted and the original images over 57 runs. The gray areas around the lines display the 98% confidence intervals. Note that these intervals are small.

MUST are more stable than the points selected by SIFT. This shows that the MUST interest points are more reliably detected over small changes of viewpoint than the SIFT interest points.
8.4. Results

Figure 8.6: The repeatability performance. The plot shows the mean repeatability proportion over the ten runs. The error bars give the 95% confidence intervals.

8.4.2 Robustness

Figure 8.5 shows the robustness results. The proportion of matching interest points between the original and the manipulated images are displayed on the y-axis. In Figure 8.5a and 8.5b, the results for the addition of noise are given. MUST is significantly less affected by the noise than SIFT. The performance of MUST is more than twice that of SIFT, making it much more robust to noise. Figure 8.5c shows the performance with respect to contrast manipulation. Although the performance of both methods is similar for the contrast enhancement, MUST shows a significantly better performance when the contrast is reduced. Also with enhanced brightness the use of MUST results in considerably better performance (see Figure 8.5d). There is no difference with reduced brightness.

8.4.3 Repeatability

The repeatability results can be appreciated in Figure 8.6. The first two bars show the repeatability results for all detected interest points, in other words when the visual buffer is not used. The repeatability of the MUST interest points is significantly higher then that of the SIFT points. The two other groups of bars display the results when the buffer is used, where the buffer parameters are optimized both per run and overall with an additional contraint that the buffer does not result in too few (< 50) or too many (> 350) landmarks per run. MUST has a slightly better result per run, whereas overall,
SIFT has a somewhat better repeatability. The differences, however, are not significant.

The overall best settings are:

<table>
<thead>
<tr>
<th></th>
<th>(N)</th>
<th>(M)</th>
<th>(\tau_1)</th>
<th>(\delta_1)</th>
<th>(\nu)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MUST</td>
<td>12</td>
<td>4</td>
<td>0.3</td>
<td>0.8</td>
<td>0.6</td>
</tr>
<tr>
<td>SIFT</td>
<td>14</td>
<td>8</td>
<td>0.3</td>
<td>0.5</td>
<td>0.0</td>
</tr>
</tbody>
</table>

### 8.4.4 Visual SLAM Performance

The SLAM performance is displayed in Figure 8.7. The results for the best settings per run show that MUST results in a significantly lower estimation error \((p = 0.05\), paired t-test) with a medium effect size, as measured by Cohen’s \(d\) \((d = .70)\). Also when the overall best settings are used, MUST performs better than SIFT with a medium effect size \((d = .61)\). The difference, however, is not significant \((p = .11\), paired t-test\), which is due to the large variance over the ten runs and the relatively small number of runs.

The overall best settings are:

<table>
<thead>
<tr>
<th></th>
<th>(N)</th>
<th>(M)</th>
<th>(\tau_1)</th>
<th>(\delta_1)</th>
<th>(\nu)</th>
<th>(\tau_2)</th>
<th>(\delta_2)</th>
<th>(\eta)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MUST</td>
<td>14</td>
<td>8</td>
<td>0.3</td>
<td>0.8</td>
<td>0.3</td>
<td>0.7</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>SIFT</td>
<td>14</td>
<td>8</td>
<td>0.3</td>
<td>0.5</td>
<td>0.8</td>
<td>0.6</td>
<td>0.7</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Not only does MUST result in better performance, it also reduces the amount of selected landmarks. MUST selects on average 40 interest points per image, and SIFT 124. These interest points are all fed to the buffer. Since the matching process is quadratic in the number of interest points, MUST greatly reduces the number of computations. Since both models are similarly fast in detecting interest points in the images, the full SLAM system is faster using MUST.

### 8.5 Discussion

In this chapter, a new interest point detector, MUST, is proposed for landmark selection based on local symmetry. The model exploits the presence of symmetrical forms in indoor environments. The interest points selected by MUST are more stable over small changes in perspective and are less sensitive to noise and changing light conditions.
8.5. Discussion

Figure 8.7: The SLAM performance. The plot shows the mean estimation error over the ten runs. The 95% confidence intervals are represented by the error bars.

than points selected on the basis of contrast by the SIFT model (Lowe, 2004). The repeatability of all selected interest points is higher for the proposed model than for SIFT if the visual buffer is not used. Most importantly, the use of symmetry results in better visual SLAM performance. This shows that exploiting symmetry results in valuable and robust landmarks for the representation of the environment. Moreover, MUST results in less interest points, which improves the processing time of the SLAM system.

There are three likely reasons for the success of using symmetry for interest-point detection. Firstly, symmetrical forms are inherently redundant. Both sides of symmetrical patterns represent the same structure. This makes a symmetrical pattern less susceptible to noise. Secondly, symmetry is a non-accidental stimulus. If there is a symmetrical pattern, it is likely that this pattern is not there by chance, but originates from something that is physical present in the environment and is thus stable and useful for SLAM. Finally, man-made environments, especially indoor environments as used in the experiments, contain many symmetrical forms, which is exploited.

The fact that repeatability drops when the visual buffer is used shows that the visual buffer does not function optimally and rejects a portion of relevant interest points. If the effectiveness of the buffer algorithm can be enhanced, the SLAM performance could be further improved. A visual buffer to select a small number of stable interest points from the high number of interest points offered by the detection methods is necessary to reduce the number of landmarks in the map.

Symmetry detection can results in the selection of complete objects. This could result
in semantically more meaningful landmarks and maps of the environment, which could be exploited, for instance, in human-robot interaction.

It can conclude that local symmetry provides robust and valuable landmarks, which result in improved visual SLAM performance.