BRISK-Based Visual Feature Extraction for Resource Constrained Robots

Daniel Jaymin Mankowitz and Subramanian Ramamoorthy

School of Informatics, University of Edinburgh, Edinburgh, EH8 9AB
daniel@mankowitz.co.za, s.ramamoorthy@ed.ac.uk

Abstract. We address the problem of devising vision-based feature extraction for the purpose of localisation on resource constrained robots that nonetheless require reasonably agile visual processing. We present modifications to a state-of-the-art Feature Extraction Algorithm (FEA) called Binary Robust Invariant Scalable Keypoints (BRISK) [8]. A key aspect of our contribution is the combined use of BRISK0 and U-BRISK as the FEA detector-descriptor pair for the purpose of localisation. We present a novel scoring function to find optimal parameters for this FEA. Also, we present two novel geometric matching constraints that serve to remove invalid interest point matches, which is key to keeping computations tractable. This work is evaluated using images captured on the Nao humanoid robot. In experiments, we show that the proposed procedure outperforms a previously implemented state-of-the-art vision-based FEA called 1D SURF (developed by the rUNSWift RoboCup SPL team), on the basis of accuracy and generalisation performance. Our experiments include data from indoor and outdoor environments, including a comparison to datasets such as based on Google Streetview.

Keywords: BRISK, BRISK0 - U-BRISK, feature extraction, localisation, resource constrained robot, *Nao* Humanoid Robot.

1 Introduction

The emergence of field robots that must persistently operate in dynamic environments brings with it the need for localisation based on features that may not have been explicitly engineered with the robot in mind. The issue is particularly problematic for resource constrained robots that must adopt a low complexity approach to computation. Generally robust localisation needs rich features such as is available from Feature Extraction Algorithms (FEAs) which form a crucial part of vision systems. FEAs are utilised in vision systems in order to detect *landmarks* (also known as *interest points*) and match them between corresponding images. FEAs can therefore be used for tasks such as localisation and are commonly used in systems such as automated-driving and underwater exploration [6,13]. Scale Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) are examples of FEAs that can be used for this task [3,5]. However, these algorithms have significant processing requirements and therefore are not applicable to a wide variety of resource-constrained systems.

S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 195–206, 2014.

Many current vision-based localisation techniques utilise stereo-vision in order to identify interest points [6, 12]. Many robot platforms, such as the Nao humanoid robot used in the Standard Platform League of RoboCup [10], do not allow for the possibility of utilising stereo vision.

In the RoboCup domain, a visual feature extraction technique termed 1D SURF, has been developed by the *rUNSWift* team for the purpose of localising Nao robots on the football pitch [2]. This method is computationally efficient but could suffer from limited accuracy when generalising to different environments, as will be illustrated in the sections to follow. To address these issues, feature extraction algorithms are required that can both generate rich features and at the same time be accurate and computationally efficient. We also focus on the single camera case, which is the current setup for the Nao humanoid robot [1].

We present a computationally efficient and accurate FEA called BRISK0 - U-BRISK which is a variant of the Binary Robust Invariant Scalable Keypoints (BRISK) FEA [7]. We show that this FEA can detect features which can be utilised for localising resource constrained mobile robots. We developed two novel matching constraints as well as a novel scoring function which is used to find the optimal parameters for FEAs. We present experiments verifying these developments and also highlight the potential for outdoor navigation using these techniques. We also outline a localisation routine incorporating the FEA as a concluding remark.

2 Algorithm Overview

Our proposed architecture of a vision-based feature extraction algorithm, as used in a localisation application, is shown in Figure 1. An image is captured by the robot's vision system (in this case, a Nao humanoid robot) and this forms the input to our FEA. Our FEA then tries to match this image to a set of stored images in an image bank. The stored images, along with their corresponding location in the environment, are manually captured by the robot prior to executing the algorithm. Descriptors for each stored image are computed and stored by the robot. The stored image whose descriptors generate the largest matching score with the input image descriptors is flagged as a match. The matching score and the stored image's corresponding geographic coordinates are passed to a localisation module which then updates the robot's position on a map. The robot will always assume a stationary position when performing this algorithm; this is analogous to a person gathering their bearings when lost in an environment.

Throughout this paper a match between two images will be referred to as an *image match*. A match between corresponding interest points between a pair of images will be referred to as an *interest point match*.

3 BRISK0 - U-BRISK

Binary Robust Invariant Scalable Keypoints (BRISK) has been recently developed by Leutenegger et al. [7]. Interest points are detected by computing a

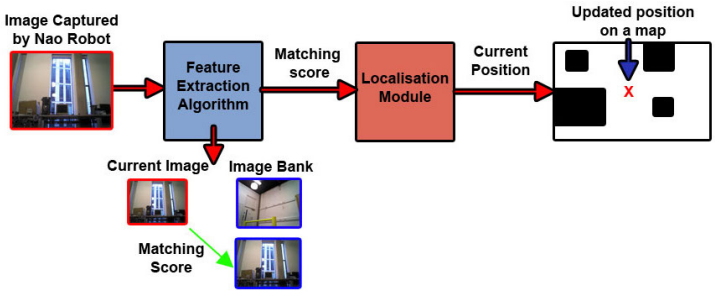

Fig. 1. The FEA incorporated with a proposed localisation module to be used by a resource constrained robot for localisation [9]

Features from Accelerated Segment Test (FAST) [11] score for each pixel in the image. If the pixel is above the pre-defined score threshold, then it is detected as an interest point. A binary feature vector of length 512 bits is then generated from some simple brightness comparison tests. Interest points are then matched between image pairs by computing the Hamming distance between the feature descriptors. This algorithm yields computational performance that betters SURF by an order of magnitude in various domains [7].

Our contribution is a modification to this original BRISK method that is more directly aimed at resource constrained robot systems with limited computation. This detector-descriptor variation is called BRISK0 - U-BRISK. BRISK0 - U-BRISK is based on SU-BRISK developed by Leutenegger et al. [7] and includes a modification to the image processing routine. This modification involves processing only a subset, such as the upper 300 pixel rows, of the captured image as this section contains more repeatable, static features such as ceiling lighting. The lower and more dynamic, less repeatable section of the image is discarded. Of course, this restriction may lose information. We compensate for this by introducing additional tests for consistency, discussed in Section 8.

3.1 BRISK0 Detector

The detector module of the BRISK0 - U-BRISK FEA is responsible for detecting the interest points. The interest points in the original BRISK implementation are invariant to both scale and rotation. To achieve scale invariance, BRISK utilises a scale-space consisting of an image pyramid whereby the lowest layer of the pyramid is the original image and the higher layers of the pyramid are down-sampled versions of the original image.

We discard the scale-space pyramid and only detect interest points on the first octave corresponding to the original image, creating a detector termed BRISK0. This is computationally efficient since down-sampled versions of the original image are discarded as well as the continuous scale refinement procedure [7]. Once all of the interest points are detected, a 512 bit descriptor vector needs to

be computed for each of the interest points. This is achieved using the U-BRISK descriptor.

3.2 U-BRISK Descriptor

In the original BRISK implementation, the descriptor vector is calculated by initially generating a pre-defined sampling pattern to sample the neighborhood surrounding the detected interest point k. The samples p_i, are equally-spaced in concentric circles surrounding the detected interest point [7]. The pattern is then rotated based on an angle α that is generated from a set of gradient calculations [7]. This is repeated for each interest point. A set of brightness comparison tests are then used to generate the 512 bit descriptor vector which is used in the matching procedure.

We do not rotate the sampling pattern, as in the original BRISK implementation, to create the U-BRISK descriptor. This prevents the FEA from being rotation invariant but improves the FEA's computational efficiency. It has been shown that the above-mentioned detector-descriptor pair is robust to rotations of up to 10° as well as scale changes of 10% or less [7]. This creates a more computationally efficient FEA which is crucial for resource constrained robots.

3.3 Image Processing

In addition to utilising a computationally efficient FEA, we implemented further optimisations. This includes optimising the 640×480 YUV image captured by the Nao robot [1]. The image is first converted to gray scale. Only interest points in the upper 300 pixels of the gray scale image are detected. These correspond to interest points near the ceiling which are less prone to changing over time. This implies that the lower portion of the captured image does not need to be processed. This results in a significant increase in computational performance.

4 Matching Feature Descriptors

Once the descriptors are computed for a pair of images, it needs to be determined whether or not the images overlap one another. We achieve this by utilising an interest point descriptor matching technique. Two techniques are utilised which include 2-NN Matching and Radius Matching.

2-NN Matching and Radius Matching both compute the Hamming or Euclidean distance between a pair of interest point descriptors in feature space for corresponding images, in order to determine whether or not the images overlap one another. The Hamming distance is computed for BRISK-based descriptors, whereas the Euclidean distance is computed for SURF-based descriptors.

As shown in Figure 1, interest points will be computed for the current image and will be matched against interest points corresponding to each image in the image bank. 2-NN Matching will compute the two closest interest point matches, ip_{match1} and ip_{match2} respectively, from an image in the image bank to

an interest point $ip_{current}$ in the current image. Only the closest matching interest point ip_{match1} is paired with $ip_{current}$. However, both ip_{match1} and ip_{match2} are required to remove invalid matches by using the well-known $2 - NN$ Ratio constraint. Radius Matching will only assign an interest point match between a pair of images if the distance between the interest point descriptors is below a pre-defined threshold. An example of matched correspondences (indicated by green lines) between two images is shown in Figure 2.

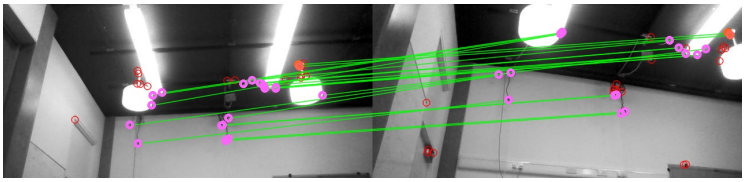

Fig. 2. Matching correspondences between two images

4.1 Matching Score

Once interest point matches are assigned to the current image, it needs to be determined whether the current image and the image in the image bank overlap one another. We achieve this by using a Matching Score (MS). The MS for a pair of interest point descriptors is calculated by taking the inverse Hamming or Euclidean distance between the descriptors [2, 4] as shown in Equation 1. For both 2-NN and Radius Matching, the inverse distance between the interest point descriptor $ip1$ in the current image and its closest match $ip2$ in an image from the image bank are used to compute the MS. Thus, interest points that are very similar will have a small distance corresponding to a high MS and vice versa. The individual MS values for every interest point is summed to produce the Image Matching Score (IMS) between a pair of images as shown in Equation 2. If the IMS is above a pre-defined threshold, then the images overlap and are flagged as an image match.

$$MS_{ip1,ip2} = \frac{1}{D_{ip1,ip2}} \tag{1}$$

$$IMS_{i1,i2} = \sum_{ip1,ip2} MS_{ip1,ip2} \tag{2}$$

For example, in Figure 2, the MS for each corresponding interest point descriptor pair (connected by a green line) is calculated. In the figure, there are 20 individual MS. These scores are summed together to produce the IMS between these images. Since these images do indeed overlap one another, the matching score between the images should be above the pre-defined IMS threshold, indicating an image match.

5 Novel Matching Constraints

For both 2-NN and Radius matching, it is important to determine whether interest point matches are indeed valid matches. Thus, we developed two novel geometric matching constraints in an attempt to remove invalid interest point matches. These constraints are termed the Angle and Distance constraints respectively.

In order to determine whether an interest point match is valid, we present an algorithm that initially places the images, containing each of the relevant interest points, next to one another as shown in Figure 3 and Figure 4. The angle constraint calculates the angle from the interest point in the left image to the interest point in the right image. Through visual analysis, it has been determined that the angle should be less than 10° in order for two interest points to match one another[1]. As can be seen in Figure 3, α is larger than the pre-defined angle threshold and therefore the interest points are invalid. β however, is smaller than the threshold and therefore a valid match (shown in green) is found.

The distance constraint states that the number of image columns separating two interest points in corresponding images should be the width of the image plus a pre-defined threshold. We have assumed small changes in rotation and scale which would again be applicable to the RoboCup domain. Through visual analysis, a pre-defined threshold of 200 pixels has been chosen. As can be seen in Figure 4, interest points that are within this threshold are flagged as valid matches (shown in green).

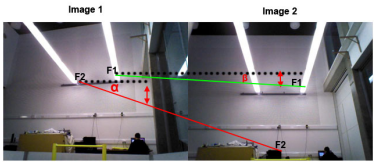

Fig. 3. The angle constraint that has been developed to remove invalid matches

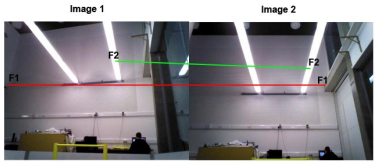

Fig. 4. The distance constraint that has been developed to remove invalid matches

6 Experimental Setup

We tested BRISK0 - U-BRISK using images captured from the upper camera of a Nao humanoid robot [1]. The Nao Robot uses an ATOM Z530 1.6 GHz processor [1]. In addition to this, the Nao's camera can process 30 frames per second. Therefore, in order to process every frame, the FEA must be able to run in under $33ms$ on the Nao's processor. All experimental image matching procedures were evaluated on a computer with an Intel Core 2 Duo T6400 2.00GHz processor[2].

[1] This is very useful for the RoboCup domain as only small rotations are expected whilst the robot is localising itself.

[2] It must be noted that the computer's processor is approximately 2.5 times faster than the Nao's processor. Therefore the computational times presented in this section are faster than the real-time performance on the Nao.

In order to ensure that the BRISK0 - U-BRISK FEA exhibits the best overall performance, we tested it against the four FEAs mentioned in Section 7. The three environments utilised for this testing procedure include a standard RoboCup environment, a large hall and an office. Examples of images captured in each of these environments are shown in Figure 5.

Fig. 5. Examples images from each of the environments used for the experiments

We generated in each environment a dataset of overlapping images and a dataset of non-overlapping images for the experiments to follow. A pair of images are considered to overlap one another if at least 50% of the first image visually overlaps the second image. This has been manually performed to ensure that the images do indeed overlap. Pairs of images with no overlap are referred to as non-overlapping images. In total 200 images were captured over all three datasets at scales varying over a range of 3 meters and orientations varying over the Nao head's full yaw range. This generated in total 2540 overlapping image pairs and 4290 non-overlapping image pairs.

7 Parameter Optimisation

In order to verify that BRISK0 - U-BRISK is a computationally efficient and accurate FEA, we compared BRISK0 - U-BRISK to variations of well known FEAs. The detector-descriptor FEAs compared include BRISK-BRISK, BRISK0-BRISK, BRISK0-SURF2D and 1D SURF as implemented by the *rUNSWift* team.

Each of these FEAs has a set of parameters that need to be pre-defined in order to accurately and efficiently match images. The parameters include the Minimum Interest Point Detection Threshold (MIPDT) which is the threshold above which a pixel is detected as an interest point; the Maximum Accepted Hamming Distance (MAHD) which is the maximum number of bits that can differ between interest point descriptors for the interests points to be flagged as a match; the Maximum Accepted Euclidean Distance (MAED) is the maximum Euclidean distance below which two descriptors are flagged as a match.

The optimal parameters $MIPDT^*$, $MAHD^*$ and $MAED^*$ for each FEA are found using a grid search and a novel scoring function that we developed for this purpose. 1D SURF uses the optimal parameter settings recommended

by the *rUNSWift* team [2] and therefore is left out in this experiment. To determine the optimal parameters, 108 training images were captured at different scales and orientations, generating a total of 1421 and 2808 overlapping and non-overlapping image pairs respectively. These images are captured by a Nao Robot that is placed on a RoboCup football pitch in a standard RoboCup environment.

The novel scoring function consists of a number of sub-functions which are now detailed. The Single Image Score (SIS) function, shown in Equation 3, represents how well a pair of images i_1, i_2 overlap one another in a particular dataset d for a particular Feature Extraction Algorithm FE and parameter values $\mathbf{p}$.

$$SIS^{FE,\mathbf{p}}_{(i_1,i_2),d} = \alpha f(t_{i_1,i_2}) + (1-\alpha) g(NVM_{i_1,i_2}) \quad 0 \leq \alpha \leq 1 \tag{3}$$

$f(t_{i_1,i_2})$, shown in Equation 4, represents the timing score for a pair of images based on the overall time (in milliseconds) taken to perform image processing, detection, extraction and matching. t_{max} is defined as the largest time tabulated for the current dataset d in milliseconds and normalises the score between 0 and 1. A large time is undesired and will result in a low matching score.

$$f(t_{i_1,i_2}) = \mid log_{10}(\frac{0.9 t_{i_1,i_2}}{t^{FE}_{max}} + 0.1) \mid \quad f(t_{i_1,i_2}) \in [0,1] \tag{4}$$

$g(NVM_{i_1,i_2})$, shown in Equation 5, rewards a pair of images if they have a large Number of Valid interest point Matches (NVM). Here, M_{total} is the total number of interest point matches between a pair of images. ϵ can be set to any value above 0 and has been pre-defined with a value of 0.1. The parameter α is a weighting parameter and takes values between 0 and 1.

$$g(NVM_{i_1,i_2}) = \frac{NVM_{i_1,i_2}}{M_{total,(i_1,i_2)} + \epsilon} \quad g(NVM_{i_1,i_2}) \in [0,1] \tag{5}$$

Once the SIS score has been calculated for each pair of images i_1, i_2, these scores are then summed together for a particular set of parameter values $\mathbf{p}$ using a specific FE in a particular dataset d. The resulting score is called the Multi-Image Score (MIS) and is shown in Equation 6.

$$MIS^{FE}_{\mathbf{p},d} = \beta\tau + (1-\beta) h(IZM) \quad 0 \leq \beta \leq 1 \tag{6}$$

The first term of the MIS score is the computation of the mean of all SIS scores for a particular set of parameter values $\mathbf{p}$, using a specific feature extraction algorithm FE, in a particular dataset d. This value is referred to as τ and is shown in Equation 7. β is a weighting parameter between 0 and 1.

$$\tau = \frac{\sum^{N}_{i_1,i_2=1,i_1 \neq i_2} SIS^{FE,\mathbf{p}}_{(i_1,i_2),d}}{N} \tag{7}$$

$h(IZM)$, defined in Equation 8, is a scoring function that accounts for the number of Image Zero Matches (IZM) for a particular set of parameter values $\mathbf{p}$ in a particular dataset d using a specific FEA. An IZM is defined as a pair of images containing no valid interest point matches. The denominator IZM^{FE}_{max} represents

the maximum number of IZMs found for a particular parameter setting in a particular dataset using a particular FEA. This function penalises parameter settings that result in a large number of IZMs since IZMs should not be present in a dataset of overlapping images.

$$h(IZM)^{FE}_{\mathbf{p},d} = \mid log_{10}(\frac{0.9IZM^{FE}_{\mathbf{p}}}{IZM^{FE}_{max}} + 0.1) \mid \quad h(IZM^{FE}_{p}) \in [0,1] \tag{8}$$

The maximum $MIS^{FE}_{\mathbf{p},d}$ is then found across all datasets and the corresponding parameters $\mathbf{p}$ for each FEA are then chosen as the optimal parameters. Using this parameter optimisation procedure, we show that BRISK0 - U-BRISK produces the best overall image matching performance.

8 Experimental Results

8.1 Comparative Performance

We found that BRISK0 - U-BRISK produced the best overall performance whilst utilising the Radius Matching technique as shown in Table 1. 1D SURF utilises the RANSAC Matching technique as developed by the *rUNSWift* team [2]. As seen in Table 1, in the worst case BRISK0 - U-BRISK can match an image pair in 12.82 ms. This is within the 33 ms time constraint required to process a pair of images for every image frame[3]. It should be noted that the times tabulated for 1D SURF utilise a sub-optimal image processing routine, which is not utilised on the *rUNSWift* robots. The times for 1D SURF can therefore be improved upon [2].

Table 1. The comparative performance for each of the FEAs in three different environments

	RoboCup		Large Hall		Office	
FEA Algorithm	AUC (%)	Time (ms)	AUC (%)	Time (ms)	AUC (%)	Time (ms)
BRISK0-BRISK	92.455	9.734	96.74	14.64	94.85	11.71
BRISK-BRISK	83.290	14.627	95.52	19.89	96.23	17.20
BRISK0-SURF2D	96.033	13.027	98.80	20.35	96.05	16.63
BRISK0-UBRISK	**97.242**	**8.805**	**93.15**	**12.82**	**96.15**	**10.45**
SURF 1D	74.039	13.301	90.84	14.03	92.33	14.14

We generated a ROC curve in the standard RoboCup environment which is shown in Figure 7. The ROC curve has been generated by utilising the IMS as the threshold which is varied from the maximum IMS in the dataset to 0.

[3] The $12.82ms$ generated by the computer has been converted to the approximate time expected on the Nao's processor. This time is still within the $33ms$ time constraint.

All overlapping image pairs with a threshold above IMS are classified as a True Positive (TP) match. False Positive (FP) image pairs are generated by non-overlapping images that are above the IMS threshold. In total we generated 1740 overlapping image pairs and 3480 non-overlapping image pairs in order to calculate the ROC curve.

The percentage Area Under the ROC Curve (AUC) for BRISK0 - U-BRISK is comparable with the other FEA methods as seen in the table. In addition, it is desirable to have a FP rate of 0 in order to prevent the robot from generating incorrect image matches. Therefore we had to determine the maximum TP rate that can be attained with a FP rate of 0, $TP^{max}_{FP=0}$, for BRISK0 - U-BRISK. As can be seen in Figure 6 BRISK0 - U-BRISK performs well in all environments attaining a minimum $TP^{max}_{FP=0}$ value of 67% in the RoboCup environment. In addition to this, it out-performs the 1D SURF routine indicating that BRISK0 - U-BRISK has superior performance in each tested environment.

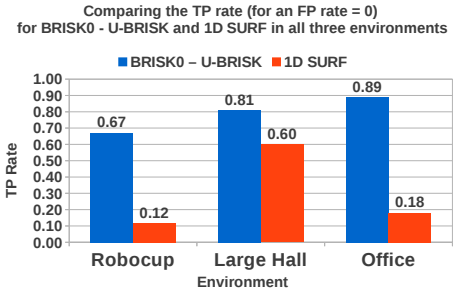

Fig. 6. The $TP^{max}_{FP=0}$ values for each of the three environments for BRISK0 - U-BRISK

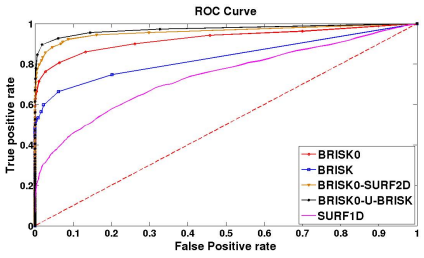

Fig. 7. A comparison of the ROC curves for the RoboCup dataset using Radius Matching

8.2 Varying Lighting Conditions

Since BRISK0 - U-BRISK has been chosen as the best overall FEA, we needed to test it under varying conditions to determine its robustness. One such test is under varying illumination. In total 5040 overlapping image pairs and 10008 non-overlapping image pairs were generated whilst switching off combinations of electrical lights in a standard indoor RoboCup environment. The image scale is fixed for this experiment. We found that the $TP^{max}_{FP=0}$ value decreased in poor lighting conditions. A minimum $TP^{max}_{FP=0}$ value of 40% resulted in the poorest lighting conditions; this implies that it is still possible to match image pairs under varying illumination albeit with poorer matching capabilities.

8.3 Outdoor Navigation

A possible application of the BRISK0 - U-BRISK FEA is utilising it on a resource constrained robot to perform navigation in an outdoor environment. Since the

algorithm is sensitive to rotation and scaling, this algorithm would be better suited to fairly stable outdoor environments. Examples include a home robot wandering around a yard, navigating around a housing complex or small suburb. In order to test this application, a dataset has been generated using Google Street View (GSV) images. If a resource-constrained robot can match its captured image I_{robot} to images that I_{robot} overlaps in the GSV dataset, I_{google}, then in principle the robot can localise itself.

For the experiment, 30 GSV images were downloaded at the same resolution as that of a Nao robot. The images were captured at latitude: 55.94474 and longitude: −3.18779. The GSV images were combined with images captured by the Nao robot at the same location. This produced 210 overlapping image pairs and 420 non-overlapping image pairs.

After running BRISK0 - U-BRISK FEA, it was found that the Nao's images can be matched to the corresponding GSV images that the Nao's images overlap, albeit with a significant decrease in matching performance. An example of interest point matches between I_{robot} and I_{google} can be seen in Figure 8. The ROC curve generated an AUC value of 77.57%. The $TP^{max}_{FP=0}$ value is 39% which implies that it is possible to match overlapping images whilst never matching non-overlapping images. One of the main reasons for the decrease in performance was due to buildings with many similar windows which caused matching ambiguities.

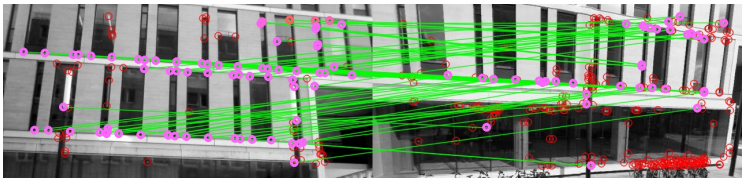

Fig. 8. Matching correspondences, shown in green, between an image captured by the Nao's camera (left image) and the Google StreetView image (right image)

9 Conclusions

We have presented BRISK0 - U-BRISK which is a unique detector-descriptor variation of the BRISK FEA and has not previously been utilised on a resource constrained robot. We found that BRISK0 - U-BRISK produced the best overall performance in terms of accuracy and generalisation to different environments. It performs significantly better than 1D SURF on these metrics. BRISK0 - U-BRISK is sensitive to changes in scale and rotation in return for computational efficiency. This may limit the applications in which the algorithm can be utilised. In harsh environments, the full power of the original BRISK implementation may be required. This FEA can still match images under varying lighting conditions albeit at a slight decrease in performance. It can potentially be used for some forms of outdoor navigation as seen in the Google Street View experiment. We

are well aware that many sophisticated techniques already exist for outdoor navigation and we do not expect our method to outdo others in terms of absolute performance. However, many scenarios involving resource constrained robots that operate in mixed environments, e.g., in the patio of a home or office, could benefit from this capability. In current work, we are looking at implementing this method as part of a particle filter localisation module shown in Figure 1.

Acknowledgements. This work has taken place in the Robust Autonomy and Decisions group, School of Informatics, University of Edinburgh. Research of the RAD Group is supported by the UK Engineering and Physical Sciences Research Council (grant number EP/H012338/1) and the European Commission (TOMSY Grant Agreement 270436, under FP7-ICT-2009.2.1 Call 6).

References

1. Aldebaran Robotics, Nao Hardware (2013), `http://www.aldebaran-robotics.com/documentation/family/robots/video_robot.html`
2. Anderson, P., Yusmanthia, Y., Hengst, B., Sowmya, A.: Robot Localisation Using Natural Landmarks. In: Chen, X., Stone, P., Sucar, L.E., van der Zant, T. (eds.) RoboCup 2012. LNCS (LNAI), vol. 7500, pp. 118–129. Springer, Heidelberg (2013)
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Computer Vision and Image Understanding 110(3), 346–359 (2008)
4. Briggs, A., Yunpeng, L., Scharstein, D.: Feature matching across 1D panoramas. In: Proc. IEEE Workshop on Omnidirectional Vision and Camera Networks (2005)
5. Juan, L., Gwun, O.: A comparison of sift, pca-sift and surf. International Journal of Image Processing (IJIP) 3(4), 143–152 (2009)
6. Lategahn, H., Geiger, A., Kitt, B.: Visual SLAM for autonomous ground vehicles. In: 2011 IEEE International Conference on Robotics and Automation (ICRA), pp. 1732–1737. IEEE (2011)
7. Leutenegger, S., Chli, M., Siegwart, R.: BRISK: Binary robust invariant scalable keypoints. In: Proc. International Conference on Computer Vision (ICCV), pp. 2548–2555. IEEE (2011)
8. Mair, E., Hager, G.D., Burschka, D., Suppa, M., Hirzinger, G.: Adaptive and Generic Corner Detection Based on the Accelerated Segment Test. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 183–196. Springer, Heidelberg (2010)
9. Mankowitz, D.J.: BRISK-based Visual Landmark Localisation using Nao Humanoid Robots. MSc. Thesis, MSc. in Artificial Intelligence, University of Edinburgh (2012)
10. RoboCup, Standard Platform League (2012), `http://www.tzi.de/spl/bin/view/Website/WebHome`
11. Rosten, E., Drummond, T.W.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006)
12. Se, S., Lowe, D., Little, J.: Vision-based Mobile Robot Localization and Mapping using Scale-Invariant Features. In: Proc. International Conference on Robotics and Automation (ICRA), pp. 2051–2058 (2001)
13. Thomas, S., Salvi, J., Petillot, Y.: Real-time Stereo Visual SLAM. MSc. Thesis, MSc. Erasmus Mundus in Vision and Robotics (2008)