Self-Calibrating Anomaly and Change Detection for Autonomous Inspection Robots

Sahar Salimpour[†], Jorge Peña Queralta[†], Tomi Westerlund[†]

[†]Turku Intelligent Embedded and Robotic Systems (TIERS) Lab, University of Turku, Finland.
Emails: [1]{sahars, jopequ, tovewe}@utu.fi

Abstract—Automatic detection of visual anomalies and changes in the environment has been a topic of recurrent attention in the fields of machine learning and computer vision over the past decades. A visual anomaly or change detection algorithm identifies regions of an image that differ from a reference image or dataset. The majority of existing approaches focus on anomaly or fault detection in a specific class of images or environments, while general-purpose visual anomaly detection algorithms are more scarce in the literature. In this paper, we propose a comprehensive deep learning framework for detecting anomalies and changes in a priori unknown environments after a reference dataset is gathered, and without need for retraining the model. We use the SuperPoint and SuperGlue feature extraction and matching methods to detect anomalies based on reference images taken from a similar location and with partial overlapping of the field of view. We also introduce a self-calibrating method for the proposed model in order to address the problem of sensitivity to feature matching thresholds and environmental conditions. To evaluate the proposed framework, we have used a ground robot system for the purpose of reference and query data collection. We show that high accuracy can be obtained using the proposed method. We also show that the calibration process enhances changes and foreign object detection performance.

Index Terms—Anomaly Detection; Change detection; Robotics; Visual anomaly detection; Computer vision; Feature extraction; Inspection robots

Fig. 1: Selection of results from the method introduced in this work.

I. INTRODUCTION

Anomaly detection, also known as foreign and outlier detection is a recurrent concept in computer vision, machine learning, and statistics. It has been explored in a wide range of research and application fields such as industry [12], medical imaging [8], security and safety systems [1]. Anomaly detectors are designed to identify the presence of unknown artifacts in data types such as images, videos, audio, text, and time series that significantly differs from the normal data. Visual anomaly detection refers to the detection of anomalies within image data. Vision-based anomaly detection can be performed on both pixel and image levels. Anomaly detection algorithms are typically based on a reference dataset from which *normal* conditions are generalized. Change detection algorithms, on the other hand, typically compare individual pairs of images to detect changes. Our objective is to combine both from the perspective of supporting autonomous inspection robots.

Machine learning methods have been extensively explored as a powerful tool for detecting anomalies. Based on reference information with lab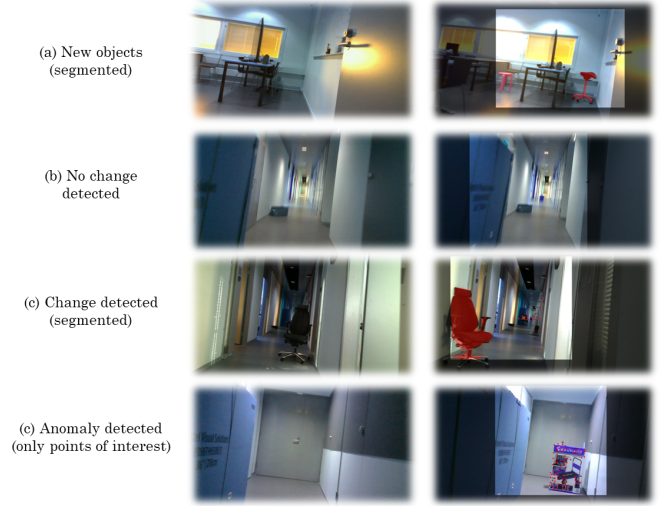elled normal and abnormal data in the training process, deep anomaly detection models can be divided into three categories: supervised, semi-supervised, and unsupervised. These anomaly detection models often require a large number of training data from both normal and abnormal data to be effective in detecting anomalies [4]. The supervised models are trained for normal and abnormal images, as well as for limited foreign objects inside abnormal images. The unsupervised models are also trained for normal data as a single-class classification, then can classify other classes as the alien classes [20].

Some of the main limitations of existing works on visual anomaly detection tasks are that they have trained their models on specific environments and datasets [30], and the models for a large variety of datasets need to have a robust localization or a very similar viewpoint [2]. Also, works that are robust against environmental changes are trained for specific environments [27].

In this study, we focus on a vision-based anomaly and change detection system for finding abnormal data based on detecting foreign or changed objects at the pixel level in single-robot missions (see Fig. 1). Anomalies or changes are both detected but not separately classified. We target the application domain of inspection robots. We first assume that the robot is able to navigate through its environment and records normal

reference images at least once. We then provide a method for self-calibration of the model based on the reference data, and a system for comparing any future images with corresponding images captured from similar locations in the reference data to detect abnormal objects or changes. The focus of our analysis has been pixel-level change detection in the second robot's camera. The evaluation was conducted on pairs of images with from different cameras and with different conditions including variable viewpoints, distance to the anomaly, and environment lightning.

When robots move in different directions and follow different routes, there can be noticeable changes in the output images, such as viewpoint changes. To overcome the mentioned challenges, a method has been applied to extract and compare multi-scale features between the input image and the corresponding image in the trusted robot. We have employed SuperPoint [5], and SuperGlue [24] for feature extraction and feature matching process between pairs of images and calibrated them based on different environments and cameras. These features are robust to lighting, scale, and viewpoint changes. Then, in order to recognize the exact object as the anomaly in the image, the trained Mask-RCNN instance segmentation model has been used to segment anomalous parts based on the extracted features. Due to the limited number of object classes in each model and the fact that there are always untrained classes, in order to address the detection of unknown anomalies, DBSAN clustering method also is applied on unmatched interest points that are not segmented. Final anomalies are detected using both the segmentation and clustering processes.

The rest of this manuscript is structured as follows. Section II discusses related work in the relevant anomaly and change detection literature. The background of the study is discussed in section III, and section IV provides an overview of the methodological approaches we used for this analysis. Results are presented in Section V, and Section VI concludes the work and points to future directions.

II. RELATED WORK

Recent studies have focused mostly on using image reconstruction approaches based on convolutional neural network models to detect visual anomalies in a certain class. Autoencoders and generative adversarial networks (GAN) are popular deep generative models. These models are trained on sufficient normal images of a single class to be able to reconstruct test images of the same class and detect the abnormal classes by comparing the loss scores [26], [35]. In [30] the authors applied a three-step procedure based on a deep generative model including an autoencoder and discriminator to detect abnormal images and foreign objects in rail images. A cognitive visual anomaly detection model consisting of an autoencoder was utilized for industrial inspection robot to detect abnormal images with larger reconstruction errors than normal images in [13].

For pixel-level anomaly detection, some studies used segmentation-based approaches. The authors in [6] developed an anomaly detection method based on the comparison of features between the input and a generated photo-realistic image using a semantic segmentation model for road images. In [9], [21] the Mask-RCNN segmentation model was used to detect anomaly events and instances for video surveillance systems. However, when anomaly is a missed object or an unknown object, it cannot be detected or segmented by the segmentation and detection methods.

Change detection is also widely used in several fields, including aerial image change detection [16], urban change monitoring [38], and anomaly detection [32] by comparing images of the same area. Wang et al. in [31] introduced a scene change detection model using the siamese vision transformer and CNN model to generate corresponding pixel-wise change maps. Similar models have been designed to be robust to outdoor conditions, including illumination, scaling, and viewpoint changes [10]. ChangeNet [29], and CSCDNet [23] which are deep siamese networks for detecting changes between pairs of images, have been used for different datasets and applications [19].

In a recent study [28], the authors presented a visual change detection method for robotics applications. They applied an attention mask to the intermediate layer of a siamese CNN to detect small changes in pairs of images. They evaluated their method by comparing reference images to live images with slight variations in viewpoint in indoor environment. A number of studies have used feature detection methods to detect anomalies in pair images [18].

There are a number of well-known classical feature detectors in the computer vision area, including SIFT [17], SURF [3], and ORB [22], as well as deep neural network-based methods, such as SuperPoint. An anomaly detection model is described in [37] using ORB features and sliding windows. In a relevant study [36], SIFT feature extractor along with polar cosine transform (PCT) have been incorporated to detect tempered pixels in images for internet of things (IoT) security. A fully-convolutional neural network architecture called SuperPoint which is a self-supervised feature point detector and descriptor was presented in [5]. SuperGlue [24] as a keypoints matching technique showed better performance in conjunction with SuperPoint to match two sets of extracted features and corresponding descriptors. The results showed that their proposed method tends to generate a larger number of correct matches that broadly cover the image compared to other traditional methods. Both SuperPoint and SuperGlue have different confidence parameters such as keypoint detectors and matching confidence thresholds which have a major effect on their performance and the final results.

III. BACKGROUND

Through this section, we introduce the key neural network architectures that are employed in our work.

A. SuperPoint

SuperPoint is a self-supervised fully convolutional neural network for extracting interest points and their descriptors.
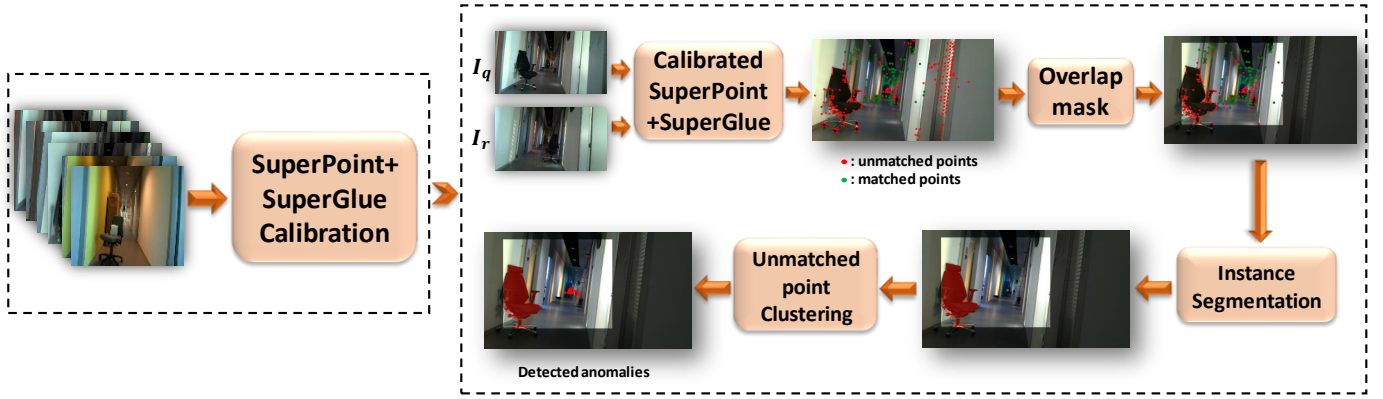
Fig. 2: Architectural diagram of the proposed framework.

Its architecture starts with pre-training a base detector called MagicPoint on synthetically generated dataset that includes simple shapes such as rectangles, stars, and cubes to extract interest points. Detecting corners of simple geometric shapes using the initial base detector performs well even with added noise [5]. However, the base detector misses many important interest point in real images. For better generalization, it is combined with homographic adaptation to provide more training samples from each image in the MS-COCO dataset. Using this approach, pseudo-ground truth interest points are generated for each image. The final network has a shared VGG-based encoder and two decoders. The first decoders is for point detection, while the second one is for point description. There are multiple shared parameters between both. Compared to traditional feature point extractors such as LIFT, ORB, or SIFT, the Superpoint model has recently achieved superior results in several research projects.

B. SuperGlue

SuperGlue is a feature point matching method that takes key points and their descriptors in image pairs and matches them with corresponding points using a graph neural network. It has been shown to produce better results in combination with the SuperPoint-generated features. In short, the algorithm is composed of two parts, an attentional graph neural network, and an optimal matching layer. A differentiable Sinkhorn algorithm is used to efficiently pair matchable points and reject non-matchable points at the final step [24]. SuperPoint and Superglue models require predefined thresholds for different parameters such as sinkhorn iterations, non-maximum suppression (NMS), and keypoints and matching thresholds. These parameters affect the performance of the networks, and therefore are often tailored to the nature of the data being fed to the models. In this work we introduce a self-calibration approach given some data samples.

C. Instance segmentation

Object segmentation is a process in which a specific class is assigned to pixel values of an image, and it is broadly divided into two types: semantic segmentation and instance segmentation. Semantic segmentation results in all pixels of the same class with the same value. Instance segmentation methods, instead, identify multiple objects of a single class as distinct instances of interest in an image, and only label pixels of classified objects. Recently, several approaches for instance segmentation have been proposed, with the most popular technique still relying on Mask R-CNN, a two-stage detection and segmentation approach [11], robust to different image types [34]. In this approach, and using an object detection model, bounding boxes are first predicted for all instances. Then, regions-of-interest (ROIs) are cropped for each instance and all ROIs are then fed to a fully convolutional network for foreground and background segmentation.

IV. METHODOLOGY

In this section, we introduce a calibration procedure for the feature extraction algorithm, along with the segmentation and clustering of interest points. A diagram of the proposed process is outlined in Fig. 2.

For detecting both changes and foreignobjects in an environment, two sets of images are processed based on the SuperPoint and the SuperGlue techniques. We calibrated the models using a single-shot procedure since the number of matched and non-matched points varies depending on different thresholds. In this study, we have used the SuperGlue GitHub repository[1] to extract and match keypoints of base and test images.

The main parameter to adjust is the SuperGlue match threshold (Δ) that directly affects the number of matched interest points. We have also analyzed the impact of other parameters but found any significant performance changes. A small Δ close to 0 (e.g., the default value $\Delta = 0.2$ used in the SuperGlue paper) results in wrong common overlap areas in image pairs. A large threshold (close to 1) , on the other hand, leads to incorrect anomaly detection. We change the amount of matching threshold from $\Delta = 0$ to $\Delta = 0.9$ with steps of 0.1. This parameter must be calibrated according to different environmental conditions and camera types. To

[1]https://github.com/magicleap/SuperGluePretrainedNetwork

this end, we capture a few image pairs with a horizontal shift with three different cameras. The images are obtained under various conditions, with changes in lightning and environment structure or background. Due to the linear shift between image pairs, we hypothesize that appropriate thresholds should yield a low standard deviation of the distribution of distances between matched keypoint pairs. Let N be the number of image pairs and mp_n be the number of matched interest points in image pair number n. The coefficient of variation (cv) of the image pairs can be computed following Eq. (1):

$$cv_n = \frac{\sigma_n}{\mu_n} \quad n = 0, ..., N \tag{1}$$

where σ_n and μ_n are the standard deviation and the average distance between matched keypoints, respectively. If the number of matched interest points is close to 0, it will result in a smaller amount of cv_n so, for the ideal threshold values with a maximum number of interest points, each cv_n is divided by the number of matched interest points. The average of this value for N image pairs is calculated as given by Eq. (2):

$$cv = \frac{\sum_{n=1}^{N} \frac{cv_n}{mp_n}}{N} \quad n = 0, ..., N \tag{2}$$

The deployment, calibration and anomaly detection process then proceeds as follows:

1. Baseline data: once a robot is deployed, we first gather a sample set of images of the new environment, which can be as small as a single pair of images. We then measured cv for different matching thresholds. Low thresholds result in more true positive, but also more false positives. The best value is a balance point between maximum number of keypoints and minimum amount of cv. We use the Kneedle algorithm to find a balance point on the curve of the mean of the coefficient of variation per matching threshold. This algorithm selects the so-called knee point. This point is defined as the furthest away from a line defined by the higher and lower points with maximum curvature [25].

2. Clean run: before starting the anomaly detection process, the robot performs a so-called *clean run* of the operational environment, which we assume to be unchanged at this state. We use this set of images as the reference for comparing any future missions, and detect anomalies

3. Image pair matching: Using onboard localization methods or other source of positioning information, relevant image pairs of two paths are selected based on both position and orientation of the robot. These images are then fed to the SuperPoint and SuperGlue algorithms with ideal threshold for matching. The positions and orientations of these image pairs are similar, but the frames aren't exactly the same, which would lead to false anomaly detections. In order to determine and process the overlap area in the query image compared to the reference image, a mask is created using the maximum and minimum matched interest points in x and y coordinates.

4. Instance segmentation: In the next step, the Mask-RCNN model is used to segment different instances in query image and the result is combined with extracted keypoints. The

---

Algorithm 1: anomaly detection in image sequences

Input:
Reference images: *r*
Reference positions: *r_p*
Query images with anomaly: *q*
Query positions: *q_p*
Range of matching and keypoint thresholds: m, k
Range of distance and orientation thresholds: d, o
Calibration:
   SuperGlue_calibration();
   DBSCAN_clusterring();
Geo_information:
   mapping();
   Geo_relevant_images();
for *r_image and q_image* **do**
   calibrated_SuperPoint_and_SuperGlue();
   Overlap_mask();
   foreach *overlapped_query* **do**
      Instance_segmentation();
      foreach *instance_mask* **do**
         n_len = length(segmented_unmatched);
         m_len = length(segmented_matched);
         if *n_len - m_len $\geq$ 1* **then**
            overlapped_unmatched.remove(
            segmented_unmatched);
            anomaly $\leftarrow$ instance_mask;
            anomaly_class $\leftarrow$ instance_class;
      DBSCAN_clusterring(overlapped_unmatched);

---

pre-trained Mask-RCNN-X101-FPN model used in this study is from the Facebook AI Research library called Detectron2, which implements most of the state-of-the-art object detection and segmentation algorithms [33]. Since this model has been trained on a huge number of COCO images, it can segment most of the probable object classes.

5. Anomalous object identification: Using segmented masks, extracted unmatched keypoints can be clustered and mapped to a specific anomalous object. Each segmented object is analyzed based on the number of matched and not matched keypoints. If the number of unmatched keypoints in an object is more, then it is considered as an anomaly. However, there are still unknown objects that cannot be segmented using any of the existing instance segmentation models, which have been trained on a wide range of classes.

In order to detect all anomalies, regardless of their classes, we apply a density-based clustering method to the unmatched interest points. At this point, we delete segmented unmatched points, and all points that remain are grouped using density-based spatial clustering of applications with noise (DB-SCAN) [7]. The DBSCAN algorithm estimates the minimum density level based on the number of neighborhood points, minPts, within a certain distance threshold, Eps. An anomaly refers to a group of unmatched interest points within this distance threshold with more than minPts neighbors. A summary of the described process can be found in Algorithm 1.
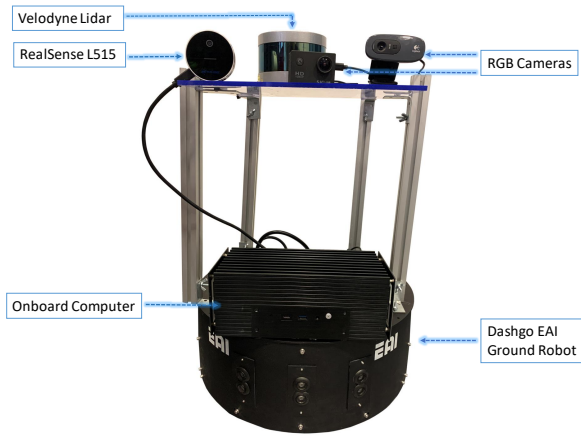
Fig. 3: Equipment utilized for data collection. The three cameras utilized in the setup have different field of view, with one of them having a wide lens and only the L515 has a global shutter. The cameras are situated in three different vertical planes to reduce the overlap between the images and increase simulate additional orientation drift for the experiments.

Fig. 4: Robot Trajectory tracking and the location of anomaly objects.

Fig. 5: Average thresholds for three different cameras for Super-Point+SuperGlue calibration.

V. Experimental Results

This section covers experiments that we carried out with a ground robot to demonstrate the effectiveness of the proposed anomaly detection and model calibration methods.

Hardware. We mounted two commercial RGB different cameras and a RealSense L515 lidar camera on a Dashgo ground robot to determine the optimum thresholds for the test environment. The robot is also equipped with a Velodyne lidar to obtain the position and orientation information using existing lidar odometry and mapping methods [14], [15]. The platform is shown in 3.

Software. The system has been implemented using ROS Melodic under Ubuntu 18.04. The robot is commanded to explore an environment and record the reference RGB images, and lidar point cloud data. Later, the robot is commanded to scan the same place in a recurrent manner to detect anomalous objects or changes. We assume that there might be slight deviations between the paths followed by the robot in consecutive iterations. Figure 4 shows the robot's trajectory during the collection of the reference data and the anomaly detection phases and the green points inside the map show different inserted anomaly objects.

Calibration. Figure 5 shows the optimum matching thresholds to calibrate SuperPoint + SuperGlue model. A few image pairs with linear shift have been recorded using three different cameras in the same environment. For each camera, average threshold values were calculated along with the minimum and maximum ranges of coefficients of variation (shaded areas) in pair images. For the RealSense L515 camera, the best value is $\Delta = 0.6$, and for the other two RGB cameras, the best value is $\Delta = 0.5$. In both cases the selected value differs significantly from the default value of $\Delta = 0.2$.

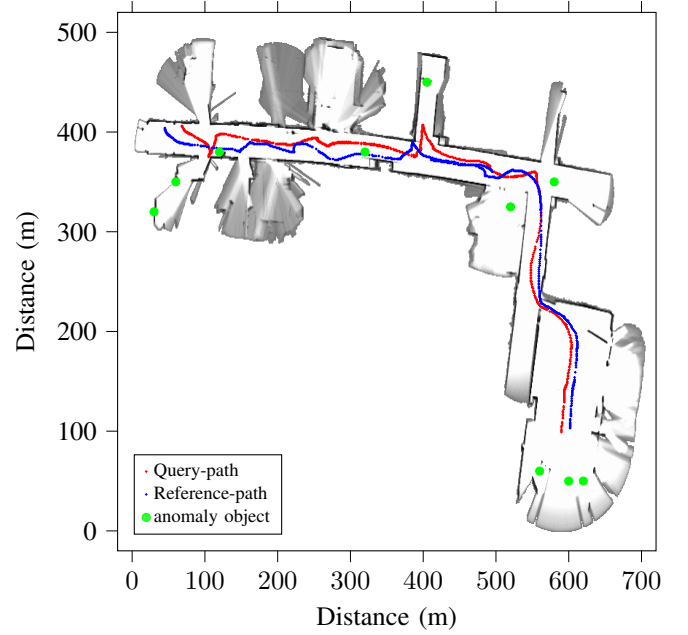It is worth noting that the DBSCAN algorithm performance depends heavily on the choice of minPts and distance threshold parameters. We have used the k-nearest neighbor's approach mentioned in the referenced paper to find the optimal distance threshold. In Fig. 7, the average Eps has been determined with a minimum of 5 neighborhood points for some random images with anomalies. Small anomalous objects cannot be clustered by selecting a larger number of minPts.

Results. When the feature detection model has been calibrated based on the optimal matching threshold, RGB images of both reference and query data are recorded using the realsense l515 camera to evaluate the proposed approach. Figure 8 shows that the proposed approach achieved 72% accuracy in detecting anomalies in more than 270 image sequences over time. In some cases, for unknown small and textureless anomalous objects that cannot also be detected

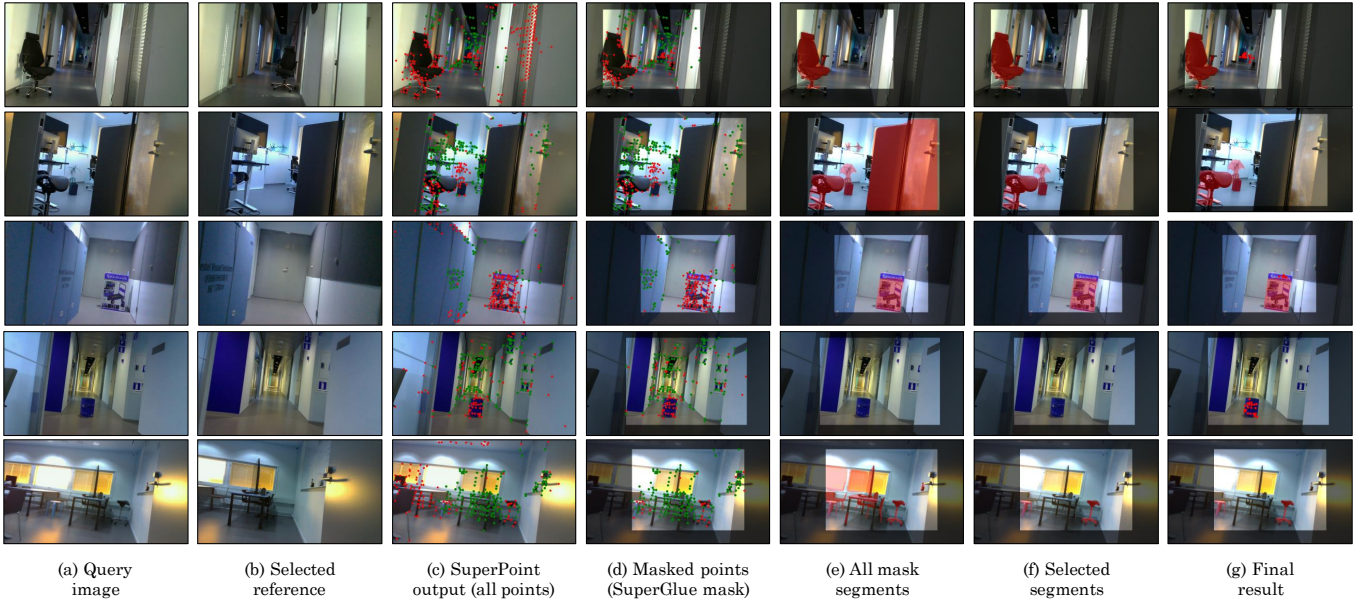| (a) Query image | (b) Selected reference | (c) SuperPoint output (all points) | (d) Masked points (SuperGlue mask) | (e) All mask segments | (f) Selected segments | (g) Final result |

Fig. 6: Selection of images in the experiment featuring different environments and types of anomaly.
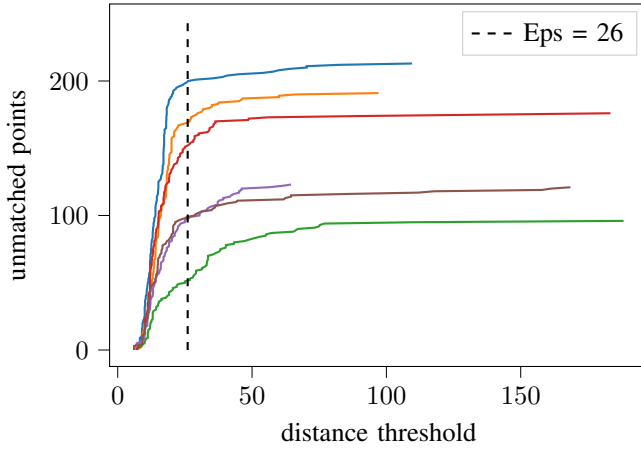

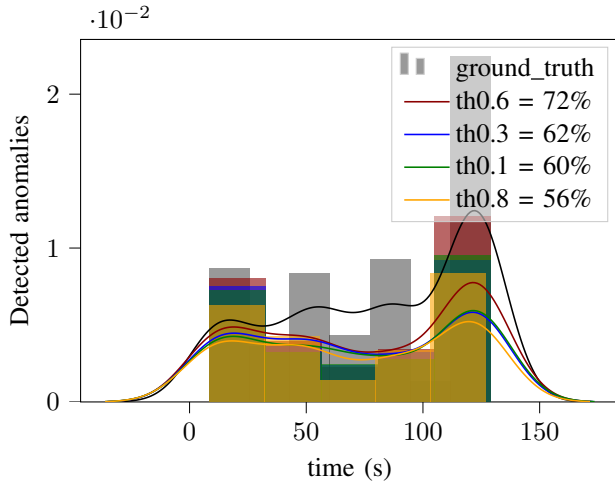Fig. 7: Average distance threshold per image.

Fig. 8: Accuracy of anomaly detection based on matching thresholds.

and segmented by the segmentation model, the proposed approach shows low accuracy for time ranges between $40\,\text{s}$ and $90\,\text{s}$ (see Fig. 8). In this figure, the performance of the proposed approach has also been analyzed for different matching thresholds. The calibrated feature extraction model with the matching threshold of $\Delta = 0.6$ achieved higher accuracy, demonstrating the effectiveness of the proposed autocalibration method. It is worth noting that the reference images are selected from the reference dataset based on the robot's position and orientation. This is a limitation as a localization error might render the method unusable. However, our method is robust to viewpoint and therefore only major errors would have an impact. In those cases, still, our method would potentially detect anomalies across the entire image, so the results can provide feedback to the localization process or robot operator indicating that there might be a large error.

Figure 6 shows a selection of the final results of the proposed anomaly and change detection workflow. The figure includes the main steps taken in the process: (a) the query image, (b) the selected reference image form the *clean* dataset based on position and orientation, (c) all the matched and unmatched points returned by SuperPoint, (d) the points that fall under the SuperGlue-generated mask, (e) the segments detected in the masked area with Mask-RCNN, (f) the selected segments where enough unmatched points are present, and (g) the segments detected as anomalies and unsegmented by clustered unmatched points that also represent potential anomalies or changes. We can see that our method works for various environmental conditions, and is able to detect anomalies and changes as segmented objects or sets of points of interest. The segmentation approach enables the detection of anomalies in largely feature-*less* objects (e.g., large flat objects such as TVs) where a clustering approach does not

work. On the other hand, the clustering of points in non-segmented image areas aids in detecting changes or anomalies regarding objects that the segmentation network is not able to classify. Overall, these two approaches together with the self-calibration show a promising way of general-purpose anomaly and change detection for mobile robots.

VI. CONCLUSION

In this paper, a visual anomaly and change detection approach for detecting pixel-level anomalies in image pairs has been presented. Using a single-shot method we have calibrated SuperPoint and SuperGlue feature detection models in order to find the proper unmatched interest points. For detecting anomalous regions and objects in the query image, a general instance segmentation model has been applied to unmatched points, as well as the DBSCAN method clustered remain points that do not belong to any object. According to the experimental results, it has been shown that this framework yields promising accuracy for a general and unknown environment and has no need for training on specific data.

In future work, we will analyze the performance across more different environments. We will also work on deploying this anomaly detection framework in a distributed manner in order to identify potentially byzantine robots within a larger multi-robot system, without a reference dataset.

REFERENCES

[1] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*. Springer, 2018.

[2] D. Avola, L. Cinque, A. Di Mambro, A. Diko, A. Fagioli, G. L. Foresti, M. R. Marini, A. Mecca, and D. Pannone. Low-altitude aerial video surveillance via one-class svm anomaly detection from textural features in uav images. *Information*, 13(1), 2021.

[3] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *European conference on computer vision*. Springer, 2006.

[4] R. Chalapathy and S. Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

[5] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *IEEE CVPR workshops*, 2018.

[6] G. Di Biase, H. Blum, R. Siegwart, and C. Cadena. Pixel-wise anomaly detection in complex driving scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.

[7] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, 1996.

[8] T. Fernando, H. Gammulle, S. Denman, S. Sridharan, and C. Fookes. Deep learning for medical anomaly detection–a survey. *arXiv preprint arXiv:2012.02364*, 2020.

[9] R. J. Franklin, V. Dabbagol, et al. Anomaly detection in videos for video surveillance applications using neural networks. In *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*. IEEE, 2020.

[10] E. Guo, X. Fu, J. Zhu, M. Deng, Y. Liu, Q. Zhu, and H. Li. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. *arXiv preprint arXiv:1810.09111*, 2018.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2017.

[12] P. Kamat and R. Sugandhi. Anomaly detection for predictive maintenance in industry 4.0-a survey. In *E3S web of conferences*, volume 170. EDP Sciences, 2020.

[13] J. Li, X. Xu, L. Gao, Z. Wang, and J. Shao. Cognitive visual anomaly detection with constrained latent representations for industrial inspection robot. *Applied Soft Computing*, 95, 2020.

[14] Q. Li, J. Peña Queralta, T. N. Gia, Z. Zou, and T. Westerlund. Multi-sensor fusion for navigation and mapping in autonomous vehicles: Accurate localization in urban environments. *Unmanned Systems*, 8(03):229–237, 2020.

[15] Q. Li, X. Yu, J. Peña Queralta, and T. Westerlund. Multi-modal lidar dataset for benchmarking general-purpose localization and mapping algorithms. *arXiv preprint arXiv:2203.03454*, 2022.

[16] R. Liu, D. Jiang, L. Zhang, and Z. Zhang. Deep depthwise separable convolutional network for change detection in optical aerial images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 2020.

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2), 2004.

[18] D. Lu, X. Liao, F. Xu, and J. Bai. Anomaly detection method for substation equipment based on feature matching and multi-semantic classification. In *2021 6th Asia Conference on Power and Electrical Engineering (ACPEE)*. IEEE, 2021.

[19] J.-M. Park, J.-H. Jang, S.-M. Yoo, S.-K. Lee, U.-H. Kim, and J.-H. Kim. Changesim: Towards end-to-end online scene change detection in industrial indoor environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.

[20] P. Perera and V. M. Patel. Learning deep features for one-class classification. *IEEE Transactions on Image Processing*, 28(11), 2019.

[21] I. V. Pustokhina, D. A. Pustokhin, T. Vaiyapuri, D. Gupta, S. Kumar, and K. Shankar. An automated deep learning based anomaly detection in pedestrian walkways for vulnerable road users safety. *Safety science*, 142, 2021.

[22] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*. Ieee, 2011.

[23] K. Sakurada, M. Shibuya, and W. Wang. Weakly supervised silhouette-based semantic scene change detection. In *2020 IEEE International conference on robotics and automation (ICRA)*. IEEE, 2020.

[24] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020.

[25] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan. Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In *31st ICDCS Workshops*. IEEE, 2011.

[26] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*. Springer, 2017.

[27] G. Slavic, M. Baydoun, D. Campo, L. Marcenaro, and C. Regazzoni. Multilevel anomaly detection through variational autoencoders and bayesian models for self-aware embodied agents. *IEEE Transactions on Multimedia*, 2021.

[28] K. Takeda, K. Tanaka, and Y. Nakamura. Domain invariant siamese attention mask for small object change detection via everyday indoor robot navigation. *arXiv preprint arXiv:2203.15362*, 2022.

[29] A. Varghese, J. Gubbi, A. Ramaswamy, and P. Balamuralidhar. Changenet: A deep learning architecture for visual change detection. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018.

[30] T. Wang, Z. Zhang, and K.-L. Tsui. A deep generative approach for rail foreign object detections via semi-supervised learning. *IEEE Transactions on Industrial Informatics*, 2022.

[31] Z. Wang, Y. Zhang, L. Luo, and N. Wang. Transcd: scene change detection via transformer-based architecture. *Optics Express*, 29(25), 2021.

[32] Z. Wang, Y. Zhang, L. Luo, and N. Wang. Anodfdnet: A deep feature difference network for anomaly detection. *arXiv preprint arXiv:2203.15195*, 2022.

[33] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[34] Y. Xianjia, S. Salimpour, J. Peña Queralta, and T. Westerlund. Analyzing general-purpose deep-learning detection and segmentation models with images from a lidar as a camera sensor. In *International Conference on Intelligent Systems Design and Engineering Applications (ISDEA), Lecture Notes in Electrical Engineering (to appear)*. Springer, 2022.

[35] V. Zavrtanik, M. Kristan, and D. Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112, 2021.

[36] W. Zhang, X. Tang, and J. Zhang. Image anomaly detection based on adaptive iteration and feature extraction in edge-cloud iot. *Wireless Communications and Mobile Computing*, 2022, 2022.

[37] X. Zhang, L. Li, J. Li, J. Lyu, R. Huang, and H. Xing. Abnormal appearance detection of substation based on image comparison. In *MATEC Web of Conferences*, volume 59. EDP Sciences, 2016.

[38] Y. Zhou, Y. Song, S. Cui, H. Zhu, J. Sun, and W. Qin. A novel change detection framework in urban area using multilevel matching feature and automatic sample extraction strategy. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, 2021.