Evaluation of Features for Change Detection in Unstructured Image Data

Friedrich Rieken Münke[1], Andreas Bartschat[1], Yujing Chen[2], Ralf Mikut[1], Markus Reischl[1]

[1] Institute for Automation and Applied Informatics
Karlsruher Institute of Technology, Karlsruhe, Germany
E-Mail: friedrich.muenke@kit.edu

[2] University of Stuttgart
Stuttgart, Germany

1 Introduction

Modern mobile devices (e.g. smartphones) are a comfortable means to gather large amounts of image data differing in place and time and can be even taken by multiple photographers. In the context of city development recently introduced deep-learning strategies make use of image data to extract objects of interest (e.g. houses, traffic, city plantation) within each image. Not only the object itself is of interest, but also its change over time, being contained in image series at other time-points (maybe taken by other users). However, images from time series of undefined imaging conditions are often not structured or systematically taken and have a high variance due to differing camera parameters (e.g. exposure, focus, positions, orientations), environment conditions (e.g. daytime, weather, light) and changes of the scene itself. Thus, the relocation of formerly found objects is challenging, even if additional GPS-coordinates are available.

In this article we evaluate features to quantify the grade of change in a detected object: Using object detection and structure from motion (SFM) techniques, objects of interest can be detected, identified and tracked over different images. With the gathered information we assess different features to describe changes over time. We discuss and evaluate features regarding consistency and fragility with respect to camera positions, lightning conditions and grade of change. Features like Scale Invariant Feature Transform (SIFT) and KAZE Features (KAZE) are designed to

be robust against different views, in this paper we discuss their suitability to detect changes of objects after the initial detection is performed with deep learning approaches.

To prove functionality, we introduce a benchmark data set that depicts an urban scene. This data set consists of unstructured images with GPS location and timestamp with a wide variety of weather conditions and camera orientations. The wide variety makes it difficult for SFM to match images. Furthermore we categorize challenges and problems forcing the algorithm to fail, such as failure of the initial object detection.

This paper serves as guideline regarding the detection of changes in unstructured image data. Assuming that a pipeline already has found objects of interest, we discuss the influences of object-changes on the representation in feature space. Therefore, we introduce the common feature-extractors SIFT, KAZE Features and Local Binary Pattern (LBP) and discuss their robustness regarding changes in objects, surroundings, image parameters etc.

We aim to:

- evaluate the performance of feature extraction strategies (SIFT, KAZE and LBP) to describe and detect changes of objects of interest in unstructured image data,
- discuss common failures in interpreting the results of feature extraction strategies if object changes occur and
- recommend best practices to cope best with object changes in unstructured image data.

In Section 2 we give a short overview about the techniques we use and the overall state of change detection. The concept of this work is described in Section 3. To evaluate the concept we build a benchmark data set, which is describe and presented in Section 4. In Section 5 the parameters and detailed processing steps are described. The results are presented and discussed in Section 6.

2 Related Work

2.1 Overview

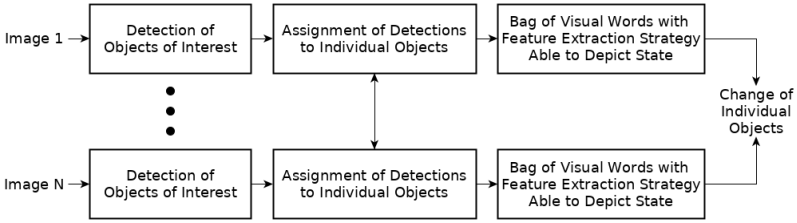
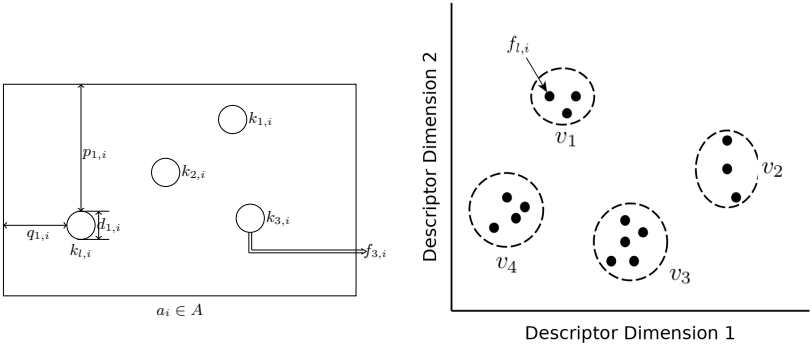


Figure 1: Pipeline to detect changes of individual objects in unstructured image data

Change detection of objects of interest in general consists of three main steps as shown in Fig. 1. First it is necessary to use state of the art object detection algorithms to segment and classify the objects of interest in all available images [13, 14]. Afterwards all detections showing the same object are grouped, utilizing e.g. structure from motion (SFM) techniques [15, 20]. This paper focuses on the last step, where we extract features which are suitable to describe the change of the objects over time. We evaluate the feature extraction strategies for monitoring the change of unique target objects. The previous steps as shown in Fig. 1 are considered done in this paper.

2.2 Feature Extraction Strategy



(a) Visualization of key points (b) Visual word dictionary creation

Figure 2: Visualization of the bag of visual words approach. a) Visualization of key points $k_{l,i} \in K_i$ and descriptors $f_{l,i} \in F_i$ on an image $a_i \in A$. b) An example of the creation of a visual word dictionary by localizing cluster centers (visual words v_w) in the set of all descriptors F. In the example the descriptors $f_{l,i}$ are $\in \mathbb{R}^2$ and the number of clusters is $N_w = 4$.

When processing a set of images $A = \{a_1, a_2, ..., a_{N_{img}}\}$ the bag of visual words approach [5] is a method to represent an image a_i as a histogram of visual words V_i. This approach is used in [6] for unsupervised texture classification and in [17] for supervised image classification. For an image a_i a set of key points $K_i = \{k_{1,i}, k_{2,i}, ..., k_{N_K,i}\}$ is defined using a key point sampling method (KSM). Each key point $k = (p, q, d)^T$ is defined by the location p, q on the image a_i in pixel coordinates and the size d of the key point as shown in Fig. 2a. For each key point $k_{l,i} \in K_i$ a corresponding descriptor $f_{l,i}$ is computed with the descriptor computation method (DCM). A descriptor $f_{l,i}$ is defined as:

$$f_{l,i} = \begin{bmatrix} x_{l,i,1} \\ x_{l,i,2} \\ \vdots \\ x_{l,i,N_{DCM}} \end{bmatrix}_{DCM} . \tag{1}$$

As result we yield a set of descriptors $F_i = \{f_{1,i}, f_{2,i}, ..., f_{N_K,i}\}$ for a given image a_i. We define the combination of key point sampling method

(KSM) and descriptor computation method (DCM) as feature extraction strategy (FES), with FES(KSM, DCM).

A visual words dictionary is created as shown in Fig. 2b. First, a target number of words N_v is selected. Then we apply k-means clustering to the set of all descriptors $F = \{F_1, F_2, ..., F_{N_{img}}\}$ to locate the N_v clusters. Every located cluster center is called visual word v_w with $w \in N_v$ and the knowledge about all located visual words is called visual words dictionary. Every computed descriptor $f_{l,i} \in F$ is assigned to one cluster center v_w.

The feature vector V_i used to represent the image a_i is built in four steps:

1. Detecting a set of key points K_i on the image a_i using the KSM
2. Computing the set of descriptors F_i for the set of key points K_i using the DCM
3. Assigning every descriptor $f_{l,i} \in F_i$ to its closest visual word v_w in the visual dictionary
4. Counting all occurring visual words to a histogram V_i

There are already feature sampling strategies (FES) implementing key point sampling methods (KSM) and descriptor computation methods (DCM). In this paper state of the art FES such as SIFT [3] and KAZE [8] are evaluated. In addition, we evaluate the DCM opponent color histograms (OCH) [4] and local binary patterns (LBP) [2]. Since OCH and LBP do not implement a KSM, we use dense sampling to generate an artificial set of key point K_i per image a_i. Dense sampling defines a fixed grid on an image a_i with fixed key point size d [9].

We propose the bag of visual words approach to detect changes in unstructured image data, since it can be used unsupervised and has proven to perform in the similar case of unsupervised image classification. The bag of visual words approach is especially suitable in this case, since the image is reduced to visual words, which are able to represent an image without encoding local information.

2.3 Dimension Reduction and Evaluation

For the evaluation of the distribution of high dimensional data points, dimension reduction methods can be utilized. In this paper we use the state of the art Uniform Manifold Approximation and Projection (UMAP) as presented in [19]. The method is presented as a general purpose method for visualization and preprocessing of data. The algorithm can be used supervised or unsupervised.

The Kullback-Leibler divergence as described in [1] is a measure to evaluate the similarity or distance of two probability distributions. The Kullback-Leibler divergences β can be computed for two sets of points $P^{\xi} = \{p_1, p_2, ..., p_{M^{\xi}}\}$ and $P^{\zeta} = \{p_1, p_2, ..., p_{M^{\zeta}}\}$, where M^{ξ} and M^{ζ} are the number of points in each set and each point $p_b \in P$ is a vector. If the set of points P is normal distributed, where μ is the center of P and Σ is the covariance matrix of P, the Kullback-Leibler divergence $\beta_{\xi,\zeta}$ for P^{ξ} and P^{ζ} is defined as:

$$\beta_{\xi,\zeta} = (\mu_{\xi} - \mu_{\zeta})^T \frac{\Sigma_{\xi}^{-1} + \Sigma_{\zeta}^{-1}}{2} (\mu_{\xi} - \mu_{\zeta}) + \frac{1}{2} \operatorname{sp}(\Sigma_{\xi}\Sigma_{\zeta}^{-1} + \Sigma_{\xi}^{-1}\Sigma_{\zeta} - 2I). \quad (2)$$

2.4 Change Detection

In this paper we divide changes in three categories: environmental, scenic and object-related. Environmental changes refer to changes in exposure, view point and angle, weather, daytime and seasons. Scenic changes are dominated by disappearing, appearing and moving objects within the scene. The last category of object-related changes describes the change of the state (size, color, shape, texture, ...) of an object. While a scenic change can be located on the image, object-related changes do not have a fixed location on the image. The following methods are trying to locate scenic changes and aim to be robust against environmental changes.

In [18] the changes of an urban environment are monitored. To recognize and detect changes in this environment a deconvolutional neural network (CDNet) was trained. The CDNet processes an image pair and marks areas with scene changes. Due to the architecture of the neural network only images which show the exact same scene can be processed and compared. Besides this limitation the network is able to detect changes and discriminate them from noise (e.g. exposure, weather, ...). The performance was evaluated on 152 sequences of images of the VL-CMU

data set. The data set was created to evaluate a self localization system. The CDNet can not be used for the use case presented in this paper since the necessary same position and orientation of the camera is not given. Due to the perspective change it is not possible to use image registration methods for compensation.

In [22] a recurrent convolutional neural network is used to analyze satellite images. Satellite images make it possible to survey a large area in defined time intervals. This end-to-end system is able to compare satellite images. The recurrent neural network encodes temporal dependencies and can reliably detect changes in the satellite images. In satellite images the distance and orientation of the camera is similar and small changes are corrected using image registration.

The Network architecture CosimNet as presented in [21] aims to detect changes in scenes with a siamese network. CosimNet is able to detect changes and segment them even with small viewpoint changes and is able to outperform CDNet. Large viewpoint changes still challenge the network. This approach focuses on scenic changes rather than changes of target objects.

All presented change detection approaches do not address object-related changes. Large viewpoint changes are still an issue for the presented change detection approaches. The topic of object-related changes is the focus of this paper. We propose a bag of visual words approach to detect object-related changes and in this paper we evaluate different feature extraction strategies to perform this task.

3 Concept

The aim is to detect changes of objects of interest O_k with $k \in N_O$ where N_O is the number of all observed objects in a set of unstructured images $A = \{a_1, a_2, ..., a_{N_{img}}\}$. Each image a_i with $i \in N_{img}$, where N_{img} is the number of all images in A, has a GPS location $l_i = (\phi, \lambda)$ in Latitude and Longitude and a corresponding time stamp t_i. The images in A can be divided in subsets A_j with $j \in N_{insp}$. These subsets A_j are called inspections, which is a series of images taken while moving through an area of interest.

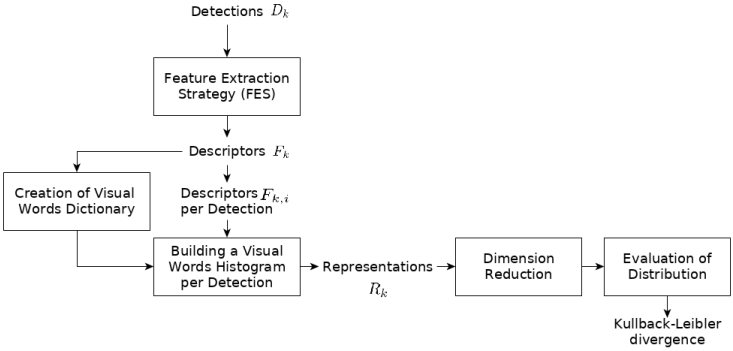


Figure 3: Evaluation of a Feature Extraction Strategy used in combination with the bag of visual words approach to detect changes in unstructured image data

The first two steps of change detection, as described in Fig. 1, are considered done. We recommend a Faster-RCNN frame work for object detection and SFM to reconstruct the view point of each image for assigning the separate detections to individual objects. At this point we have a set of bounding boxes (detections) D. Each bounding box (detection) $d_{k,i}$ shows an object O_k and is part of an image a_i. All bounding boxes (detections) $d_{k,i}$ assigned to one object O_k are called a set of detections $D_k = \{d_{k,1}, d_{k,2}, ..., d_{k,N_k^d}\}$, where N_k^d is the number of detections of object O_k. Since this paper focuses only on the comparison of feature extraction strategies to depict object-related changes, we assume, that the set of detections D_k for each object O_k is complete and that each detection $d_{k,i}$ is assigned to the correct object O_k. An object O_k has a set of unique states $S_k = \{s_{k,1}, s_{k,2}, ..., s_{k,N_{k,S}}\}$, where $N_{k,S}$ is the unknown number of states of the object O_k. Every detection $d_{k,i}$ shows the object O_k in a state $s_{k,z} \in S_k$, fuzzy-states are not considered in this paper. With the bag of visual words approach we extract a feature vector from a detection $d_{k,i}$ as representation $r_{k,i}$ of the state $s \in S_k$. The parameters of the process are defined in Section 5. We define the set of representations R_k of an object O_k as $R_k = \{r_{k,1}, r_{k,2}, ..., r_{k,N_k^d}\}$. A change of an object O_k from State s_ξ to State s_ζ should be noticeable in the distribution of the sets of representations R_k^ξ and R_k^ζ of both states. This paper focuses on the performance of different feature extraction strategies (FES) used in the bag of visual words approach. The FES are evaluated separately for different objects O_k. The evaluation process of one FES is shown in

Fig. 3. The bag of visual words approach (Section 2.2) is applied to all objects O_k with all feature extraction strategies (FES) to be evaluated. In the first step the FES extracts a set of descriptors $F_{k,i}$ per detection $d_{k,i}$. The set containing all available descriptors is called F_k. The visual words dictionary is created with F_k, while each representation $r_{k,i}$ is build using $F_{k,i}$. The dimensions of each representation $r_{k,i} \in R_k$ is reduced to $\hat{r}_{k,i} \in \mathbb{R}^2$ with UMAP for visual assessment. In the end the FES performance is evaluated using the Kullback-Leibler divergence between all representations $\hat{R}_k$ grouped by their states $s \in S_k$.

To the best of our knowledge, we are the first to compare the performance of SIFT, HSV-SIFT, KAZE, HSV-KAZE, OCH and LBP in a bag of visual words approach to detect object-related changes of an object O_k. The performance is tested considering differing viewing points, viewing angles, exposures, weather, daytime, motion blur and different types of object-related changes.

4 Dataset

There are many data sets dedicated to change detection [7, 11, 12]. They mostly focus on the scenic changes, where the appearance, disappearance and movement of objects are the most dominant changes. The data sets features different seasons and daytime, while having fixed or only slight changing viewpoints for each scene.

This paper focuses on object-related changes, where the state of an object itself changes. Additionally the change detection needs to be robust against environmental changes especially for different viewpoints. As a result the location of the change on the image is not important, since the viewpoint could have changed and the location of the change not necessarily reflects a change of the object state $s \in S_k$.

(a) Image: 222
24.06.19, 07:37:33

(b) Image: 223
24.06.19, 07:37:36

(c) Image: 226
24.06.19, 07:37:45

Figure 4: Sample images from inspection A_{10}

Figure 5: Example annotation of the benchmark data set A

To evaluate the performance of feature extraction strategies (FES) to separate the states $s \in S_k$ of an object of interest O_k, we introduce a benchmark data set A with $N_{img} = 1121$ and $N_{insp} = 38$. The benchmark data set A covers 300m of a street in Karlsruhe, Germany. The first inspection A_1 was collected on the 16.05.2019 and the last inspection A_{38} was collected on the 12.08.2019. Every inspection A_j is a series of images, which was collected by taking pictures in random intervals while walking down the street. Therefore an inspection A_j has constant weather, daytime and general viewing direction. Each image a_i has a resolution of 4032px $\times$ 3024px. In addition to the constant external influences during an inspection A_j the images $a_i \in A_j$ have different view points and viewing directions. Therefore the objects O_k are viewed from different angles, distances and are influenced by exposure, weather and daytime. Sample images are shown in Fig. 4. The only meta-information provided is the GPS signal from the smartphone and the time stamp. As

ground truth different states $s \in S_k$ of the objects of interest and their bounding box (detection) $d_{k,i}$ on the image are annotated, as seen in Fig. 5. It is assumed that the state $s \in S_k$ of each object O_k remains constant for each inspection A_j. The state of each object is annotated in a separate text file.

In the data set we have three objects of interest O_1, O_2 and O_3. The first object O_1 is a flower with two states $S_1 = \{0, 1\}$, where 0: "Blooming" (Fig. 6a) and 1: "Faded" (Fig. 6b). The second object O_2 is a rose with the states $S_2 = \{0, 1\}$, where 0: "Blooming" (Fig. 6c) and 1: "Not Blooming" (Fig. 6d). The third object O_3 is a poster stand with the states $S_3 = \{0, 1, 2, 3\}$, where 0: "Seniorenkino" (Fig. 6e), 1: "Film x Musik" (Fig. 6f), 1: "Hollywood" (Fig. 6g) and 2: "Met Opera" (Fig. 6h). All objects changed significantly during the time of observation.

(a) Blooming Flower (b) Faded Flower (c) Blooming Rose (d) Not Bloom. Rose

(e) Poster Seniorenkino (f) Poster Fim x Musik (g) Poster Hollywood (h) Poster Met Opera

Figure 6: All observed objects of interest in the data set

The object of interest O_1 "flower" is compact. The information about the object is focused in the center of the bounding box. The observed change refers to the color difference, while the texture itself remains constant.

The second object of interest O_2 "rose" is not as compact and spread out in the bounding box. The change refers to the appearance of red blossoms. The last object of interest O_3 "poster stand" changes with different posters displayed in the same stand. In this case the bounding boxes show nearly no background. The change itself is dominant and features different colors and textures.

5 Methods

5.1 Preprocessing

The preprocessing consists of the first two steps mentioned in Fig. 1. For object detection any deep learning object detection framework (e.g. Faster RCNN) can be used. The performance of object detection algorithms depends on the type of object and the trainings dataset.

The detections $d_{k,i}$ are assigned to individual objects O_k in two steps. First, we use SIFT key point matching and the image GPS coordinates to reconstruct the view point and view angle of every image in 3D coordinates. Afterwards the true position of each detection can be estimated by projecting it into the 3D space. Detections which share the same location are assigned to a common object. With this method we can assure that even after significant changes all detections of an object are assigned to the same object.

5.2 Image Preprocessing

Since each detection $d_{k,i}$ has a different scale, due to differing distances of view point of image a_i to the object O_k, all detections $d_{k,i}$ are resized to a fixed size of 300px $\times$ 300px. This is useful if the observed object O_k does not change its actual size during the period of observation. Afterwards the detection $d_{k,i}$ is standardized to reduce the effects of exposure and daytime. The preprocessed detection $d_{k,i}$ is then processed by the bag of visual words approach (Section 2.2).

Depending on the requirements of the feature extraction strategy (FES) the detection $d_{k,i}$ is transformed from RGB color space to gray scale $d_{k,i}^{gray}$ or HSV color space $d_{k,i}^{hsv}$ as described in [4].

5.3 Representation of the Object

As described in Section 3 the bag of visual words approach is used to build a representation $r_{k,i}$ for each detection $d_{k,i}$. While the feature extraction strategies are evaluated and changed, the creation of the visual words dictionary and the building of the histogram as representation is constant for all tests. We use k-means clustering to create the visual words dictionary and to build the histogram of visual words. We have empirically chosen $N_v = 400$ as size of the visual words dictionary.

Abbr.	FES	KSM	DCM
sift	FES(SIFT, SIFT)	SIFT	SIFT
kaze	FES(KAZE, KAZE)	KAZE	KAZE
hsv-sift	FES(SIFT, HSV-SIFT)	SIFT	HSV-SIFT
hsv-kaze	FES(KAZE, HSV-KAZE)	KAZE	HSV-KAZE
och	FES(DENSE, OCH)	DENSE	OCH
lbp	FES(DENSE, LBP)	DENSE	LBP

Table 1: The specifications of all tested feature extraction strategies (FES)

We evaluate six feature extraction strategies FES(KSM, DCM): *sift*, *kaze*, *hsv-sift*, *hsv-kaze*, *och* and *lbp*. The specifications of each FES are shown in Tab. 1. SIFT and KAZE both provide a native key point sampling method (KSM) and a native descriptor computation method (DCM). Both work with the processed detection $d_{k,i}^{gray}$. HSV-SIFT and HSV-KAZE are a specialisation of the basic SIFT and KAZE descriptor computation method (DCM) which is applied to the preprocessed detection $d_{k,i}^{hsv}$. OCH and LBP are descriptor computation methods without a key point sampling method. OCH uses $d_{k,i}$ to compute descriptors and LBP computes descriptors based on $d_{k,i}^{gray}$. As key point sampling method we have chosen dense sampling, where a fixed grid of key points is defined. The grid parameters were chosen empirically with a step size of 10px and a key point size of 15px.

5.4 Evaluation of Feature Extraction Strategy Performance

The representations $\hat{R}_k$ of an object O_k are built as described in Section 3. $\hat{R}_k$ contains representations $r_{k,i}$ each with a defined state $s \in S_k$, which was annotated in the benchmark dataset (Section 4). A FES is suitable

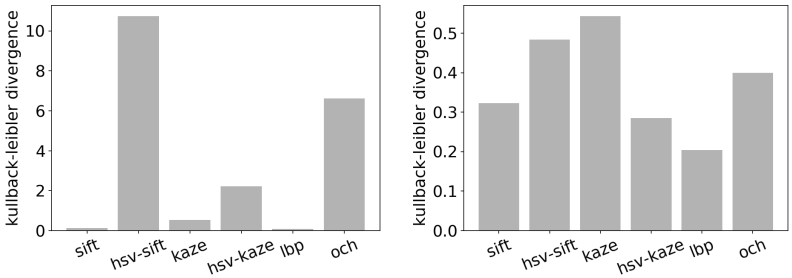
to detect changes of an object O_k, when the representations of each state can be separated in feature space. The Kullback-Leibler divergence $\beta_{\xi,\zeta}$ is used to measure the distance between two sets of representations $\hat{R}_k^\xi$ and $\hat{R}_k^\zeta$ with the corresponding states s_ξ and s_ζ. We assume that $\hat{R}_k^\xi$ and $\hat{R}_k^\zeta$ are normal distributed and compute $\beta_{\xi,\zeta}$ as described in Section 2.3. When an object O_k has more than two states the Kullback-Leibler divergence is calculated for each state pair and the mean over all divergences is used measure the performance.

6 Results

All feature extraction strategies FES (Tab. 1) were tested on the three chosen objects of interest O_1, O_2 and O_3. The performance of all FES is evaluated as described in Section 3 and the results are shown in Fig. 7. The Kullback-Leibler divergence β of each FES is evaluated per object. Depending on the object O_k the interval of β varies. A small β represents a strong similarity for two sets of representations $\hat{R}_k^\xi$ and $\hat{R}_k^\zeta$. This is not desirable, since the change detection relies on the separation of both sets.

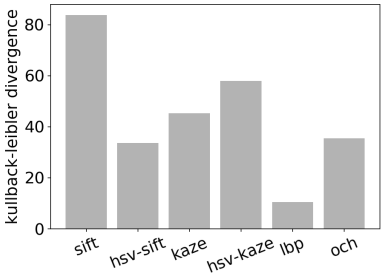
The β of the first object O_1 "flower" is shown in Fig. 7a. For object O_1 *hsv-sift* and *och* reach the highest β with $\beta_{hsv-sift} = 10.7$ and $\beta_{och} = 6.1$. As shown in Fig. 8 both FES are able to separate both states of object O_1. The change of the object O_1 is mainly color-related, as expected FES considering colors (*hsv-sift* and *och*) perform best. The FES *hsv-kaze* performs worse with $\beta_{hsv-kaze} = 2.2$. In the visual evaluation *hsv-kaze* still shows tendency to separate both states. All FES (*sift*, *kaze*, *lbp*) which do not consider color information, are not able to make a separation and therefore are not able to detect the change between State 0 and State 1.

In Fig. 7b the Kullback-Leibler divergences β of all tested FES regarding the second object O_2 "rose" are presented. The highest β reached for object O_2 is $\beta_{kaze} = 0.5$. The visual evaluation confirms that no FES was able to separate the two states 0: “Not Blooming” and 1: “Blooming” from each other. Object O_2 is due to the sparse information in the bounding box and the small changes especially difficult. In Fig. 9 the distribution of all representations $\hat{R}_2$ is shown.



(a) Kullback-Leibler divergence for all tested FES for O_1 (flower)

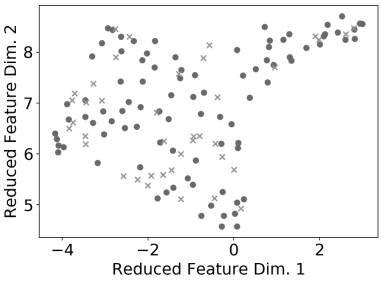
(b) Kullback-Leibler divergence for all tested FES for O_2 (rose)



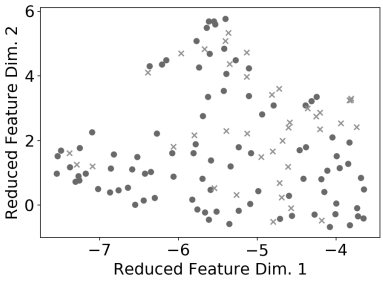
(c) Kullback-Leibler divergence for all tested FES for O_3 (poster stand)

Figure 7: The performance to separate the states $s \in S_k$ of an object O_k measured in Kullback-Leibler divergence β of all tested feature extraction strategies (FES) for each object O_k

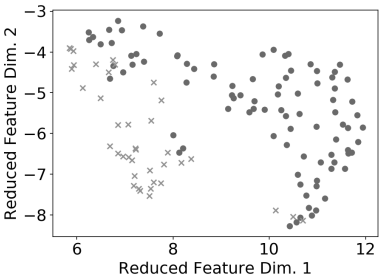
The Kullback-Leibler divergences β of the object O_3 are shown in Fig. 7c. O_3 has four states 0: Seniorenkino, 1: Film x Musik, 2: Hollywood and 3: Met Opera. Every state has dominant texture and color characteristics, as result all FES are able to separate the states from another to some degree. The minimum β is $\beta_{lbp} = 10.5$ and the maximum is $\beta_{sift} = 83.8$. The visual evaluation in Fig. 10 confirms that all FES are able to separate the state.



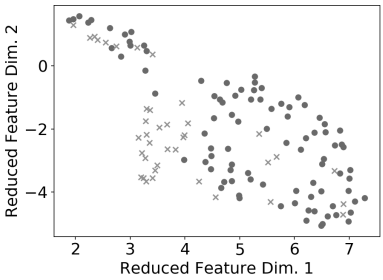
(a) Feature Extraction Strategy: *sift*



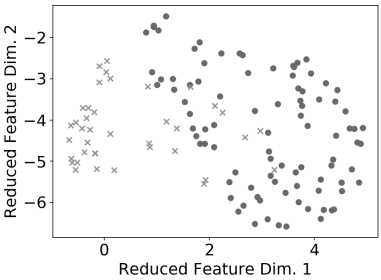
(b) Feature Extraction Strategy: *kaze*



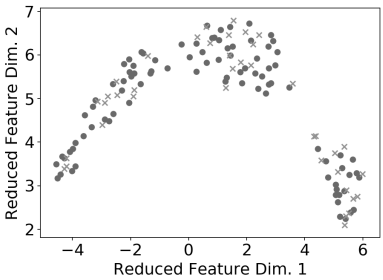
(c) Feature Extraction Strategy: *hsv-sift*



(d) Feature Extraction Strategy: *hsv-kaze*

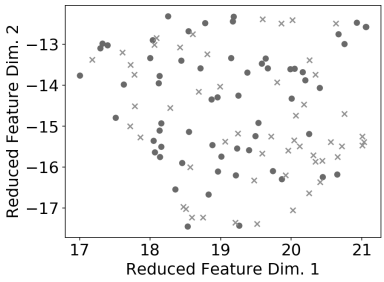


(e) Feature Extraction Strategy: *och*

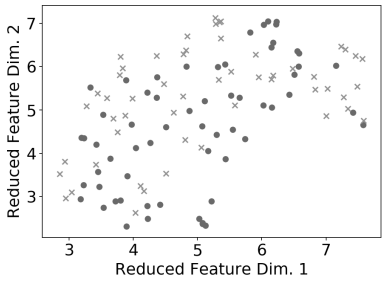


(f) Feature Extraction Strategy: *lbp*

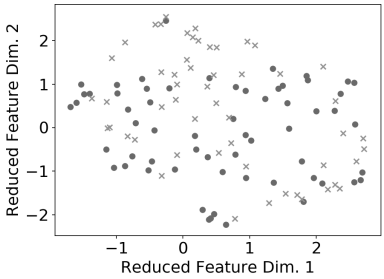
Figure 8: Distribution of representations $\hat{R}_1$ of the object O_1 (flower) in feature space [Circle: “Blooming”, Cross: “Faded”]



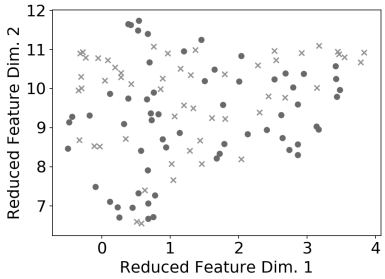
(a) Feature Extraction Strategy: *sift*



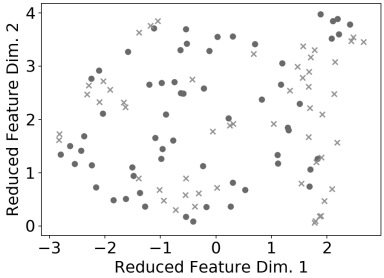
(b) Feature Extraction Strategy: *kaze*



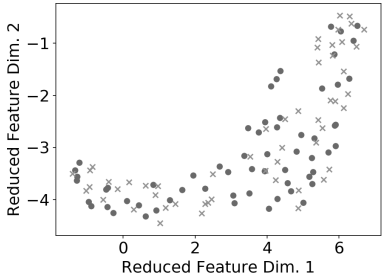
(c) Feature Extraction Strategy: *hsv-sift*



(d) Feature Extraction Strategy: *hsv-kaze*

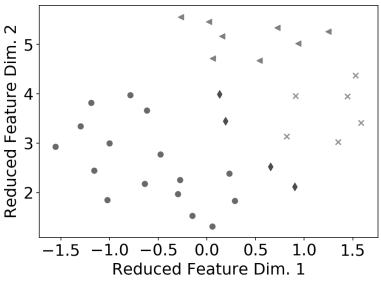


(e) Feature Extraction Strategy: *och*

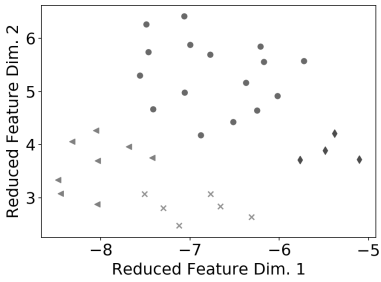


(f) Feature Extraction Strategy: *lbp*

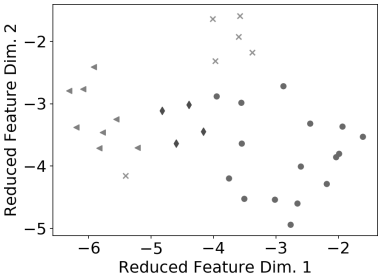
Figure 9: Distribution of representations $\hat{R}_2$ of the object O_2 (rose) in feature space [Circle: "Blooming", Cross: "Not Blooming"]



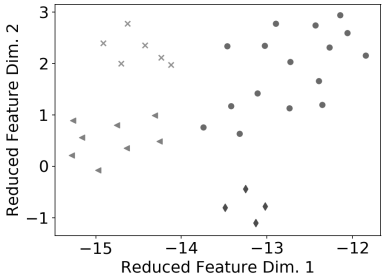
(a) Feature Extraction Strategy: *sift*



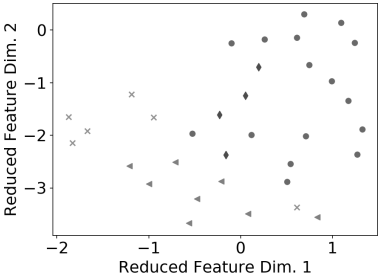
(b) Feature Extraction Strategy: *kaze*



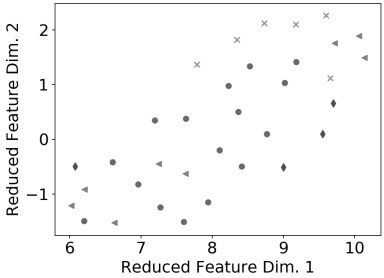
(c) Feature Extraction Strategy: *hsv-sift*



(d) Feature Extraction Strategy: *hsv-kaze*



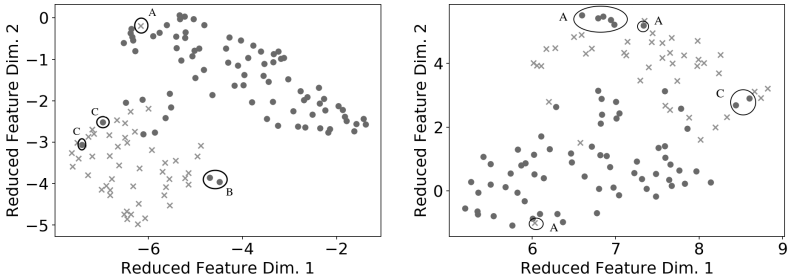
(e) Feature Extraction Strategy: *och*



(f) Feature Extraction Strategy: *lbp*

Figure 10: Distribution of representations $\hat{R}_3$ of the object O_3 (poster stand) in feature space [Circle: “Seniorenkino”, Cross: “Film x Musik”, Triangle: “Hollywood”, Diamond: “Met Opera”]

We tested the other FES combining different KSM and DCM. The FES(KAZE, SIFT) had the noticeable ability to separate the viewing/walking directions from each other. The distributions of representations R_1 and R_2 of the objects O_1 and O_2 are shown in Fig. 11. The two viewing directions (North and South) are clearly separated for both objects. The outliers in Fig. 11 were evaluated. Group A are detections of occluded objects. Group B are detections, which were taken while passing the object. This leads to a differing viewing and walking direction. Group C represents images which were taken from across the street.



(a) Representations R_1 of object O_1 (b) Representations R_2 of object O_2

Figure 11: Distribution of representations R_k created with FES(KAZE, SIFT) marked by viewing direction [Circle: North, Cross: South]

Further evaluations to measure the performance on new validation data is planed, such as analyzing the classification performance to separate different states.

7 Conclusion

In this paper we evaluate the suitability of the bag of visual words approach for object-related change detection. The results indicate that the approach is viable and able to handle unstructured image data. Overall the FES *hsv-sift*, *hsv-kaze* and *och* have proven to be robust against environmental changes and capable to detect different object-related changes. The FES *lbp* performed worst for all objects. The combination of the independent FES *kaze* and *och* are able to detect

changes of an object O_k and give information whether the change is color or texture related. This combination is suitable for a wide variety of changes.

Since the change of object O_2 could not be detected, more tests with new FES are needed. Test related to other color transformations, other DCM and optimization of the parameters (e.g. number of words, clustering-method, ...) of the visual words dictionary could further improve the performance. In addition more robust representations could be built from many detections instead of one. For future projects the visual words dictionary could be evaluated to gain insights in the meaning of changes.

References

[1] S. Kullback and R. Leibler "Information and Sufficiency" In *Annals of Mathematics and Statistics* 22:79–86, 1951.

[2] T. Ojala, M. Pietikäinen and T. Mäenpää "Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns" In *European Conference on Computer Vision, LNCS* 1842:404–420, 2000.

[3] D. Lowe "Distinctive Image Features from Scale-Invariant Keypoints" In *International Journal of Computer Vision* 60(2):91–110, 2004.

[4] S. Sergyan "Color Content-based Image Classification" In *Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence and Informatics*, 5:427–434, 2007.

[5] J. Yang, Y. Jiang, A. Hauptmann and C. Ngo "Evaluating Bag-of-Visual-Words Representations in Scene Classification" In *MIR Proceedings of the International Workshop on Multimedia Information Retrieval*, 7:197–206, 2007.

[6] L. Qin, Q. Zheng, S. Jiang, Q. Huang and W. Gao "Unsupervised Texture Classification: Automatically discover and classify Texture Patterns" In *Image and Vision Computing* 26(5):647–656, 2008.

[7] H. Badino, D. Huber and T. Kanade "Visual Topometric Localization" In *Intelligent Vehicles Symposium*, 2011

[8] P. Alcantarilla, A. Bartoli and A. Davison "KAZE Features" In *European Conference on Computer Vision, LNCS* 7577(4):214–227, 2012.

[9] T. Tuytelaars "Dense Interest Points" In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2281–2288, 2013.

[10] J. Kasecka and G. Mason "Detecting Changes in Images of Street Scenes" In *Computer Vision - ACCV*, 11(4):590–601, 2013.

[11] Y. Wang, P. Jodoin, F. Porikli, J. Konrad, Y. Benezeth and P. Ishwar "CDnet 2014: An Expanded Change Detection Benchmark Data Set" In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014

[12] K. Sakurada and T. Okatani “Change Detection from a Street Image Pair using CNN Features and Superpixel Segmentation” In *BMVC* 61:1–12, 2015

[13] S. Ren, K. He, R. Girshick and J. Sun “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks” In *arXiv: 1506.01497v3*, 2016

[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu and A. Berg “SSD: Single Shot MultiBox Detector” In *arXiv: 1512.02325v5*, 2016

[15] J. Schönberg and J. Frahm “Structure-from-Motion Revisited” In *Conference on Computer Vision and Pattern Recognition*, 2016

[16] L. N. Thin, L. Y. Ting, N. A. Husna and M. H. Husin “GPSSystems Literature: Inaccuracy Factors and Effective Solutions” In *The International Journal of Computer Networks & Communications (IJCNC)* , 8(2):123–131, 2016.

[17] A. Bartschat, J. Stegmaier, S. Allgeier, K. Reichert, S. Bohn, O. Stachs, B. Koehler and R. Mikut “Augmentations of the Bag of Visual Words Approach for Real-Time Fuzzy and Partial Image Classification” In *Workshop Computational Intelligence, Dortmund*, 27:227–242, 2017.

[18] P. Alcantarilla, S. Stent, G. Ros, R. Arroyo and R. Gherardi “Street-View Change Detection with Deconvolutional Networks” In *Autonomous Robots*, 42:1301-1322, 2018.

[19] L. McInnes, J. Healy and J. Melville “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction” In *The Journal of Open Source Software*, 3(29):861, 2018.

[20] P. Gargallo, O. Lorentzon, Y. Noun and Y. Noutary “OpenSFM Mapillary” *https://github.com/mapillary/OpenSfM 2018.*

[21] E. Guo, X. Fu, J. Zhu, M. Deng, Y. Liu, Q. Zhu and H. Li “Learning to Measure Changes: Fully Convolutional Siamese” Metric Networks for Scene Change Detection In *arXiv: 1810.09111v3*, 2018.

[22] L. Mou, L. Bruzzone and X. Zhou “Learning Spectral-Spatial-Temporal Features via a Recurrent Convolutional Neural Network for Change Detection in Multispectral Imagery” In *IEEE Transactions on Geoscience and Rémote Sensing*, 57(2)924–935, 2019.