OMNIDIRECTIONAL OBJECT DUPLICATE DETECTION

Peter Vajda, Ivan Ivanov, Lutz Goldmann, Touradj Ebrahimi

Multimedia Signal Processing Group – MMSPG
Institute of Electrical Engineering – IEL
Ecole Polytechnique Fédérale de Lausanne – EPFL
CH-1015 Lausanne, Switzerland

ABSTRACT

In this paper, we extend a graph-based approach for omnidirectional object duplicate detection in still images. Objects are detected from several points of view with different distances. The goal of this work is to determine how many training images have to be taken and from which points of view in order to achieve a certain efficiency. Moreover, the performance of the algorithm is improved by automatically generated images, where the original training images are scaled and rotated in 3D space. Our experiments show that four training images are enough for 3D object duplicate detection from a planar view point and ten training images for omnidirectional detection.

***Index Terms*—** object duplicate detection, graph matching, SIFT, omnidirectional detection, visual search

1. INTRODUCTION

With the technological evolution of digital acquisition and content analysis, millions of images and video sequences are captured every day and used in a large variety of applications. As keyword-based indexing is very time consuming and inefficient due to linguistic and semantic ambiguities, content-based image and video retrieval systems have been proposed, which search and retrieve documents based on the content itself rather than its associated tags or keywords. Within such systems, a query document is usually compared to all the documents in a database through visual features extracted from it. However, since the features are extracted from images which contain two-dimensional projections of three-dimensional objects, the features may change significantly depending on the view point. Thus, systems could fail to retrieve relevant content in response to some queries.

In general, content-based image retrieval can utilize different low-level representations for describing the image content, such as global descriptors, regions or feature points. Recently, interest has turned towards higher-level representations such as object-based. Given a query image containing an object, an image retrieval task can be reformulated as an object duplicate detection task. The goal of the object duplicate detection is to detect the presence of a target object in an image based on an object model created from training images. Duplicate objects may vary in their perspective, have different sizes, or could be modified versions of the original object after minor manipulations, which do not change their identity. Therefore, object duplicate detection should be robust to changes in position, size, view point, illumination, and partial occlusions.

A large number of applications can benefit from a precise object duplicate detection. For example, when a user takes a picture of an object with his/her mobile phone, additional information about the object can be retrieved from the web, such as the price of a product, or the name and location of a monument. Moreover object duplicate detection may be used to search a specific object in a large collection, such as a suspect car in a video surveillance database. In this case, objects should be detected from any view point and at any size with a certain efficiency. Therefore it is important to understand the limits of omnidirectional object duplicate detection which is the focus of this paper.

We analyze an earlier proposed graph-based approach [1] for 3D object duplicate detection in still images, considering detections from any view point and with any scaling factor. A graph model is used to represent the 3D spatial information of the object based on features extracted from the training images so that a complex 3D object processing is avoided. Therefore, improved performance can be achieved in comparison to existing methods in terms of robustness and computational complexity. The main goal is to determine how many images of the object of interest should be captured in order to detect it with a certain precision. We also analyze the positions of the cameras from which the images should be captured in order to reach the optimal (minimal) number of training images. Furthermore, we show how synthetic training images can be created through an affine transformation in order to decrease the number of captured training images. A database is created and used for an in-depth analysis of omnidirectional object duplicate detection. The database contains images of several object classes taken from different points of view and different distances.

The remaining sections of this paper are organized as follows. We introduce related work in the next section. Then, we describe our approach for object duplicate detection in more details. Next, experiments and results are shown. Finally, we conclude the paper with a summary and perspectives for future work.

2. RELATED WORK

Typically, most object duplicate detection methods contain the following steps: feature extraction, object representation, and matching. In this section we review representative object duplicate detection methods based on these steps.

Local features are used for object duplicate detection in [2]. The General Hough transform is then applied for object localization. Furthermore, posture of the object with respect to the camera is estimated using the RANSAC algorithm. Our object duplicate detection method is based on this algorithm and the detection accuracy is improved by using a spatial graph matching method. This method is also extended by considering more training images. Therefore, 3D objects can be detected with higher accuracy. In [3], descriptors are extracted from local affine-invariant regions and quantized into visual words, reducing the noise sensitivity of the matching operation. Inverted files are used to match the video frames to a query object and retrieve those which are likely to contain the same object. However, this work considers only 2D objects, such as posters, signs, ties, and does not take into account real 3D objects. In this paper, an analysis of this method for real 3D objects is provided. An extension [4] of this approach uses key-point tracking to retrieve different views of the same object and to group video shots based on the object's appearance. The tracked object is then used as an implicit representation of the 3D structure of the object to improve the reliability of object duplicate detection. This method has proven to be more effective when compared to a query with a single image, but it requires that all the relevant aspects of the desired object are present in the query shot, which limits its applicability.

Most of the object representations consider the objects in the 2D image space only. But, since real-world objects are inherently 3D, a higher performance can be achieved using 3D models. However, the creation of complete 3D models requires a large number of images from all possible angles, which may not be feasible in real applications. Despite this difficulty, interesting solutions have been proposed for multi-view retrieval of objects from a set of images or video. In [5] a full 3D model of the object is used for the detection of objects in video sequences. Our approach makes an attempt towards 3D modeling, while keeping the efficiency of 2D processing, using a graph model to represent the 3D spatial information.

Different scale and orientations of the objects can be evaluated for better performance of the feature extraction and the salient region detection tools. Mikolajczyk analyzed different affine region detectors in [6] considering different angles and distances. However region detectors are just one small part of an object duplicate detection method, feature descriptors are also necessary for local feature matching. In the original papers of SIFT [2] and SURF [7] feature descriptors, their effectiveness is analyzed for different view points and scales. However, the performance of the algorithms is evaluated for a very limited number of planar objects. In [8], the author goes a step further and creates new features which include several affine transformed generated images as input. This paper shows significant improvement over the original SIFT description. As we will show later, the number of generated images can be decreased, using optimal camera positioning. We also use a significantly larger database with more than 80 different objects. Moreover, we describe an optimal strategy for training image creation for full omnidirectional detection.

3. OBJECT DUPLICATE DETECTION

The goal of object duplicate detection is to detect the presence of a target object in an image, based on an object model created from training images. We make an attempt towards 3D modeling, while keeping the efficiency of 2D processing. A graph model is used to represent the 3D spatial information of the object based on the features extracted from training images so that we can avoid explicitly building a complex 3D object model. Therefore, improved performance can be achieved in comparison to existing methods in terms of robustness and computational complexity.

3.1. Training phase

Training is performed as follows: given a set of images, local features are extracted and a spatial graph model describing the object is created. First, regions of interest (ROIs) in an image are detected using the Hessian affine detector [9] and SIFT features [2] are extracted from each region. These features are robust to arbitrary changes in viewpoints. Then, hierarchical k-means clustering [10] is applied to the features, to group them based on their similarity. The result of the hierarchical clustering is used for a fast approximation of the nearest neighbor search, in order to speed up the local feature matching. Finally, a spatial graph model is constructed to improve the accuracy of the feature matching, which considers scale, orientation, position and neighborhood of features. The nodes of the graph are the features of training images. The edges of the graph connect features with their spatial nearest neighbors. The attributes of edges are the distance and orientation of the neighbors. These attributes are important for the matching step in the test phase.

3.2. Testing phase

To detect the presence of a specific object in a test image, features are extracted from the image in the same way as described before. These features are matched to those in the graph model derived from training images using a one-to-one nearest neighbor matching. Considering only matched features and their positions, a spatial graph model of the query image is constructed in the same way as described in the training phase. Then, graph matching is applied between the two graph models to identify the local correspondences between the local features in the training and the test image. Finally, for the global object matching and matching score computation, the general Hough transform is applied on the nodes of the matched graph. The matching scores represent the pair-wise comparison of training and test images.

More details about the proposed object duplicate detection approach are provided in [1].

4. EXPERIMENTS

Imagine a scenario in which object duplicate detection is used to search for a specific object, such as a suspect car, in a video surveillance database. In this case, objects should be detected from any view point and any size with a certain efficiency. Then an interesting question is how many training images of the object are necessary to detect it with a certain accuracy. And moreover, is it possible to decrease the number of captured images by generating synthetic images using affine transformations?

4.1. Synthetic training images

One way to improve the accuracy of the object duplicate detection algorithm is to generate synthetic images using affine transformations on the original training images. To generate synthetic images we scaled the original images by s^n, where $n \in [0 \dots 10]$ and s is a parameter that was set to 0.85 in our experiments. Rotation does not distort the image linearly with the angle, therefore synthetic rotated images are generated by scaling the training image in the horizontal direction by s^n. In object detection we consider only one direction of rotation, assuming that the results for other directions do not change much. All generated images are used as training images for object duplicate detection in the experiments.

4.2. Database

The experiments were based on an object database which contains 850 images. This database was created in order to evaluate the object duplicate detection method from different view points and distances. It consists of ten 3D and five 2D object classes as shown in Figure 1: bag, bicycle, body, face, shoes, stone, can, car, building, motor, poster, logo, newspaper, book and workbook. Each of the 15 classes contains at least three different objects, and together 85 objects are contained in the database with ten sample photos per object.

Figure 2 shows three images for two selected classes: building and shoes. As it can be seen from these samples, images with a large variety of view points ($0° - 90°$ spread equally) and sizes ($1 - 0.15$ relative size spread equally) are considered for each class. The angles and relative sizes of each object are calculated for each image in the dataset to facilitate the analysis of the omnidirectional duplicate detection as shown in Section 4.4.

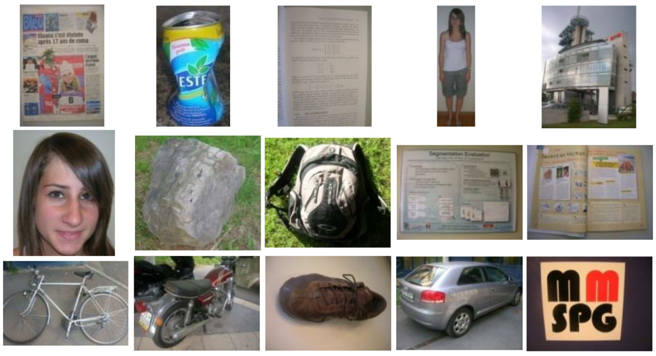

Fig. 1. Samples of the different 2D and 3D objects classes in the database used.

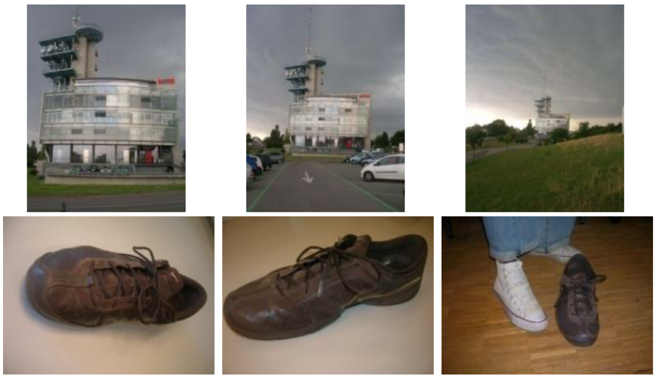

Fig. 2. Image samples for two objects under diverse viewing conditions in the database used.

4.3. Evaluation

Object duplicate detection can be evaluated as a typical detection task [11] using correspondences between a set of predicted objects, and a set of ground truth objects. Each image of our database contains just an object, therefore we are not evaluating the locations of the objects, but just their presence. A pair-wise comparison of ground truth and predicted objects is then performed. The results are used to obtain the values of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). The resulting confusion matrix serves as a basis on which different curves can be derived.

The precision recall (PR) curve plots the precision (P) versus the recall (R) with:

$$P = \frac{TP}{TP+FP} \quad (1)$$

$$R = \frac{TP}{TP+FN} \quad (2)$$

This curve does not consider TN which is not uniquely defined for detection problems.

In order to determine the optimum thresholds for object detection, the F-measure is calculated as the harmonic mean of P and R values, given by:

$$F = \frac{2 \cdot P \cdot R}{P+R} \quad (3)$$

which considers P and R equally weighted. Optimizing this value can resolve the threshold selection problem. However in the case of the surveillance scenario, recall is more important when compared to precision.

4.4. Results and analysis

Imagine a scenario where a surveillance system should detect a knife, a gun, a stolen bag or any other suspicious object. For a reliable system, the object duplicate detection should achieve a certain F-measure, even if these objects are shown from an arbitrary direction. In the following, examples of 0.7 and 0.8 F-measures are selected arbitrarily for illustration purpose.

It is very difficult to create a system which can detect every object with a precision higher than a certain level. For example it is impossible to detect a paper if the training image was taken from its edge. Therefore we cannot guarantee this for every object. Our database contains 15 classes of objects with high diversity. Therefore, we assume that a particular object in a given scenario can be detected with high enough accuracy, for some angle and scale factors.

In this section, we evaluate the performance of the proposed object duplicate detection algorithm with respect to angle and size deviations between training and test images in order to derive requirements for omnidirectional object duplicate detection. Furthermore, we explore the benefit of synthetic training images generated through affine transformations.

The results of the analysis are shown in Figures 3 and 4. Using only the original training images, the F-measure starts to decrease considerably when the object size in the test image is less than 60% of the original size, or when the viewing angle differs by more than $40°$ from the training image. When the viewing angle differs by $90°$, the F-measure drops to 0.45. In contrast to previous research, this shows that real objects have to be considered as convex 3D objects rather than planar 2D objects which could lead to an increased tolerance to angle deviations. Small and distorted objects on the query images create results with high variance below 30% of relative size, as depicted in Figure 3. Adding synthetic training images generated by affine transformations leads to a significant improvement of the F-measure (up to 0.2) over the whole range of size deviations. However, for the angle deviations the F-measure improves significantly (up to 0.15) only for angles larger than $50°$. These results are expected since the scaling of the training images needed for different sizes causes much smaller distortions in the synthetic images than the affine transformations required for the different angles.

Based on these results, it is possible to derive the minimum number of training images and the required angles and distances of the objects in the images from our database in order to achieve a certain overall F-measure value. In order to achieve an F-measure of at least 0.8 by using an object model trained with only one training image, the test images may dif-

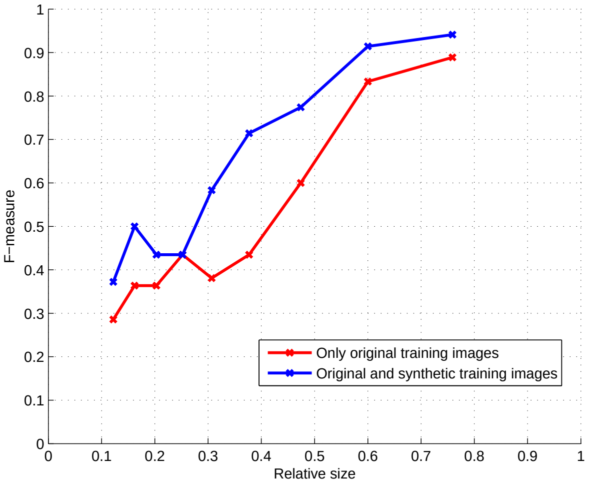

Fig. 3. F-measure vs. relative size of the object in the test image.

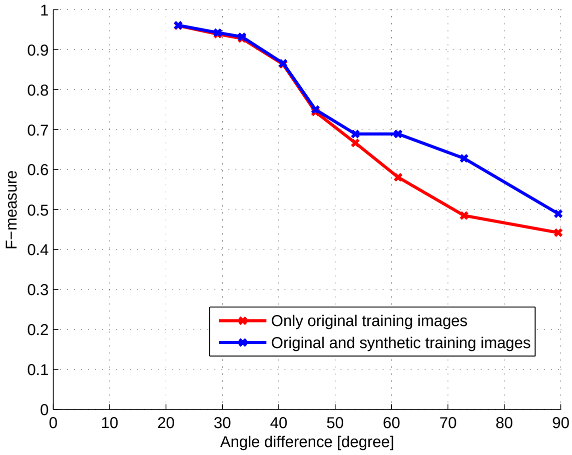

Fig. 4. F-measure vs. viewing angle difference between the training and test images.

fer from the training image up to an angle of ±45° and up to a size of 59%. Therefore, if we want to detect at least 80% of the test objects for all possible rotations around a single axis in the given scenario, four training images are enough because one image can cover 90° of 360° as shown in Figure 5. Using synthetic training images, the scale factor improves from 59% to 50% for an F-measure of 0.8, while the angle difference does not change.

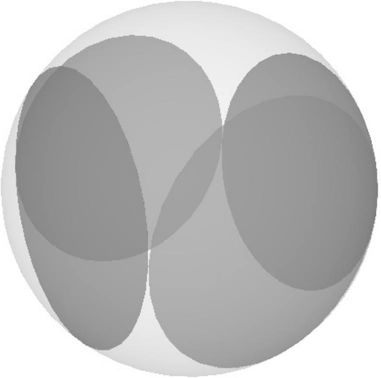

Fig. 5. Figure shows the suggested camera positions for planar object duplicated detection with 0.8 F-measure, where each disk represents a camera and its coverage area.

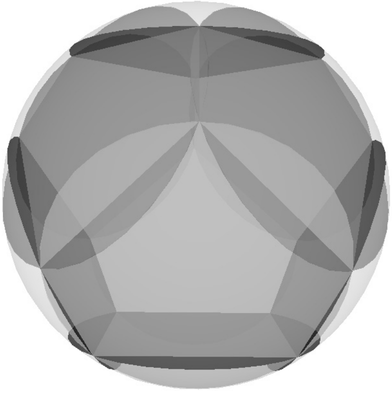

Fig. 6. Figure shows the suggested camera positions for omnidirectional object duplicated detection with 0.8 F-measure, where each disk represents a camera and its coverage area.

If we consider omnidirectional object duplicate detection in the 3D space, it is necessary to solve the problem of positioning disks (or, equivalently, cameras) to cover a sphere. More precisely, the problem is to find the minimum number of congruent disks that cover a sphere for a given radius of the disks, or conversely, to find the minimum radius of the disks to cover a sphere for a given number of disks so that every point of the sphere belongs to at least one disk. Although a general solution of the problem for an arbitrary number of disks is not available, the solutions for some cases has been given by Fejes Tóth [12]. For different numbers of cameras, the required coverage radius of the cameras is shown in Table 1 [13].

Table 1. Solutions for the problem of covering a sphere with $\#cameras$ congruent, overlapping disks. The second row shows the radius of the disks in degree. Each disk can represent a camera and its coverage angle.

$\#cameras$	4	5	6	7	8	9
radius $[degree]$	70.53	63.43	54.74	51.03	48.14	45.88

$\#cameras$	10	11	12	13	14	15
radius $[degree]$	42.31	41.43	37.38	37.07	34.94	34.04

Table 2. Ten 3D coordinates of the centers of the disks which cover a unit sphere when the radius of the disks are 45°.

Axis/Cam	1	2	3	4	5
x	-0.521	0.449	-0.577	-0.526	0.526
y	0.576	0.879	0.684	-0.345	0.345
z	-0.630	0.160	0.446	0.778	-0.778

Axis/Cam	6	7	8	9	10
x	0.468	-0.904	0.957	-0.013	0.142
y	0.072	-0.357	-0.290	-0.594	-0.970
z	0.881	-0.236	-0.015	-0.805	0.198

Therefore, to cover a sphere with disks having a radius of 45°, 10 training images are enough, if the positioning of the cameras is as shown in Figure 6, where radius of 45° is assumed to achieve at least 0.8 for F-measure. The positions of the cameras in this case are shown in Table 2. Figure 7 combines the previous results and shows how many training images in our scenario are necessary for a certain F-measure using automatically generated syntectic images. If we would like to detect at least 70% of the test objects contained in images taken from any direction, based on the previously discussed estimations, it is allowed to have angle differences up to 50° and thus it is sufficient to use 8 images for training. However if we use synthetic training images, seven images are enough as shown in Figure 7.

5. CONCLUSION

Image and video retrieval systems are becoming increasingly important in many applications. Video surveillance and semantic image or video search are among some of the appli-

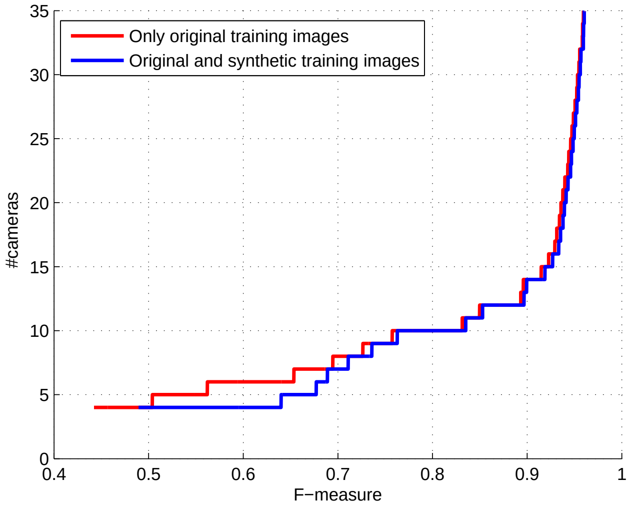

Fig. 7. F-measure vs. number of cameras needed for omnidirectional object duplicate detection using only original or additionally synthetic training images.

cations which require accurate and efficient omnidirectional object duplicate detection methods. In this work, we have extended our robust graph-based object duplicate detection algorithm for 3D objects. A novel methodology of determining the number of training images is presented in this paper. Assuming that a specific object is detected with high enough precision, for some angle and scale factors, the following conclusions can be drawn from our experiments:

- Four training images are enough for 3D object duplicate detection from planar view point.
- Eight and ten training images are necessary for full omnidirectional detection by keeping F-measure above 0.7 and 0.8 respectively.
- Synthetic training images improve mainly the accuracy of omnidirectional detection for poor performing cases, however they improve significantly the detection from different distances in all case.

As future work, we will explore omnidirectional object duplicate detection considering the dependency between angle difference and relative size.

6. ACKNOWLEDGMENTS

This work was supported by the Swiss National Science Foundation Grant ”Multimedia Security” (number 200020-113709), the Swiss National Foundation for Scientific Research in the framework of NCCR Interactive Multimodal Information Management (IM2), and partially supported by the European Network of Excellence PetaMedia (FP7/2007-2011).

7. REFERENCES

[1] Peter Vajda, Lutz Goldmann, and Touradj Ebrahimi, “Analysis of the limits of graph-based object duplicate detection,” in *Proceedings of the International Symposium on Multimedia*, 2009, pp. 600–605.

[2] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] J. Sivic and A. Zisserman, “Video Google: Efficient visual search of videos,” in *Toward Category-Level Object Recognition*, J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, Eds., vol. 4170 of *LNCS*, pp. 127–144. Springer, 2006.

[4] Josef Sivic, Frederik Schaffalitzky, and Andrew Zisserman, “Object level grouping for video shots,” *International Journal of Computer Vision*, vol. 67, no. 2, pp. 189–210, 2006.

[5] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce, “Segmenting, modeling, and matching video clips containing multiple moving objects,” in *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, 2004, pp. 914–921.

[6] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, “A comparison of affine region detectors,” *International Journal of Computer Vision*, vol. 65, no. 12, pp. 4372, 2005.

[7] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, “Surf: Speeded-up robust features,” in *Proceedings of the 9th European Conference on Computer Vision*, 2006, pp. 404–417.

[8] Jean-Michel Morel and Guoshen Yu, “Asift: A new framework for fully affine invariant image comparison,” *SIAM J. Img. Sci.*, vol. 2, no. 2, pp. 438–469, 2009.

[9] Krystian Mikolajczyk and Cordelia Schmid, “An affine invariant interest point detector,” in *Proceedings of the 7th European Conference on Computer Vision (ECCV 2002)*, 2002, pp. 128–142.

[10] D. Nister and H. Stewenius, “Robust scalable recognition with a vocabulary tree,” in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, 2006, pp. 2161–2168.

[11] Tom Fawcett, “An introduction to ROC analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[12] L. Fejes Tóth, *Lagerungen in der Ebene auf der Kugel und im Raum*, Springer-Verlang, Berlin, 2nd edition, 1972.

[13] R. H. Hardin, N. J. Sloane, and W. D. Smith, “Spherical coverings,” 1997, Retrieved from `http://www.sphopt.com/math/question/covering.html`.