The European Journal of Research and Development

journals.orclever.com/ejrnd

OR CLEVER Science & Research Group

Research Article

Embedding of Regional Adjacency Graph in Textile Image Classification with Deep Learning Application

Ömer Akgüller [1], Mehmet Ali Balcı[2], Aysu İldeniz [3*], Duygu Yavuzkasap Ayakta [4]

[1] Muğla Sıtkı Koçman University, Mathematics Department, 0000-0002-7061-2534, oakguller@mu.edu.tr
[2] Muğla Sıtkı Koçman University, Mathematics Department, 0000-0003-1465-7153, mehmetalibalci@mu.edu.tr
[3] YUNSA, Research and Developement Department, 0000-0001-8299-9433, aildeniz@yunsa.com
[4] YUNSA, Research and Developement Department, 0000-0003-0108-9839, textilengineer.duygu@gmail.com
[*] Correspondence: aildeniz@yunsa.com

(First received March 01, 2022 and in final form June 06, 2022)

Abstract

The image classification problem is a process that many machine learning methods are trying to solve. Graphs, which are combinatorial mathematical structures, are frequently used in machine learning problems. In this study, a method using machine learning based embeddings of weighted regional graphs for image classification problem is proposed.

Keywords: Laplacian Matrix, Regional Graph, Image Classification, Textile Engineering

1. Introduction

The contribution of artificial intelligence to the industrial field has been much more than expected (Babina et al., 2020; Bolton et al., 2018). The development and application areas of artificial intelligence in the industrial field, the work of many engineers and researchers, are increasing. With Germany's determination of the Industry 4.0 framework in 2010, this framework was accepted among the member states of the European Union and started to be used. Later, the United States and China determined their plans and policies for smart production and shared them with the public (Thoben et al., 2017).
Industrial AI, which combines data science, network analysis and information security, is often used to detect defects that arise from process operation and mechanical problems. Defects reduce product quality, negatively affect customer satisfaction and cause losses for the business (Bullon et al., 2017; Perez et al., 207). In addition, the error that occurs at any stage in the fabric production process negatively affects the next production stages.

https://doi.org/10.56038/ejrnd.v2i2.71

Therefore, it is very important for the enterprise to be able to detect errors in the production process effectively and quickly (Hanbay et al., 2016).
Manual control of fabric production errors in the textile industry reduces the production speed and increases the cost. The automatic control of this control is very important for increasing the quality and reducing the cost in the textile industry (Song et al., 2020). Detection algorithms are very important for the automation of detection of production defects. Many researchers have made extensive studies in recent years for the development of efficient detection algorithms (Mahajan et al., 2009).

2. Materials and Methods

In this study, local contrast is determined and salience is calculated by considering the neighborhoods of an image region at various scales. This process is the distance between the average feature vector defined on the pixels of an image subregion and the average feature vectors of neighboring pixels. Thus, instead of combining different salience maps for the scalar values of each feature, a feature map is obtained by using a feature vector.
For a fixed scale, the contrast-based saliency value c_ij can be calculated as follows: where D is Euclidean distance for uncorrelated vector elements and Mahalanobis distance for correlated elements, N_i are the pixel numbers of i-th region, and v_* are feature vectors of respected region.

$$c_{ij} = D\left(\left(\frac{1}{N_1}\sum_{p=1}^{N_1} v_p\right), \left(\frac{1}{N_2}\sum_{q=1}^{N_2} v_q\right)\right)$$

where D is Euclidean distance for uncorrelated vector elements and Mahalanobis distance for correlated elements, N_i are the pixel numbers of i-th region, and v_* are feature vectors of respected region.
Graph theory provides a tool for mapping data structures and finding relationships between data objects and allows the application of statistical techniques and machine learning.

Promising research avenues in this field are interactive visual data mining combined with graph-based data analysis. Another benefit of graph-based data structures is the application of methods from network topology, network analysis, and data mining. The application of graph theory to image analysis has been the focus of research some time ago and still presents many challenges and still requires a new approach. Graphs are main mathematical object because many pattern recognition problems are formulated through graphs, which are complex combinatorial objects that can model relational and semantic information in data.

One of the most common graph structures used in image processing is region adjacency graphs. Region adjacency graph (RAG) is a mathematical structure that encodes adjacent regions of an image. A region r_i is defined by quadruple

$$R_i = (k, x_k, y_k, A_k),$$

where k is the index number, (x_k, y_k) is the center of gravity and A_k is the area. In Figure 1, we present an example for RAG.

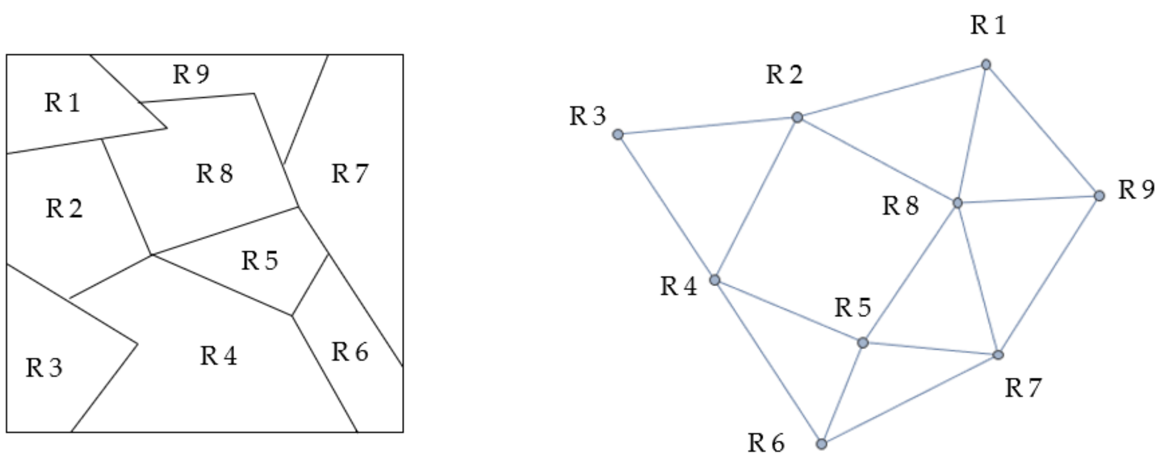

Figure 1 An example for RAG

In Figure 2, a microscopic fabric image having 100 regions and its RAG is presented.

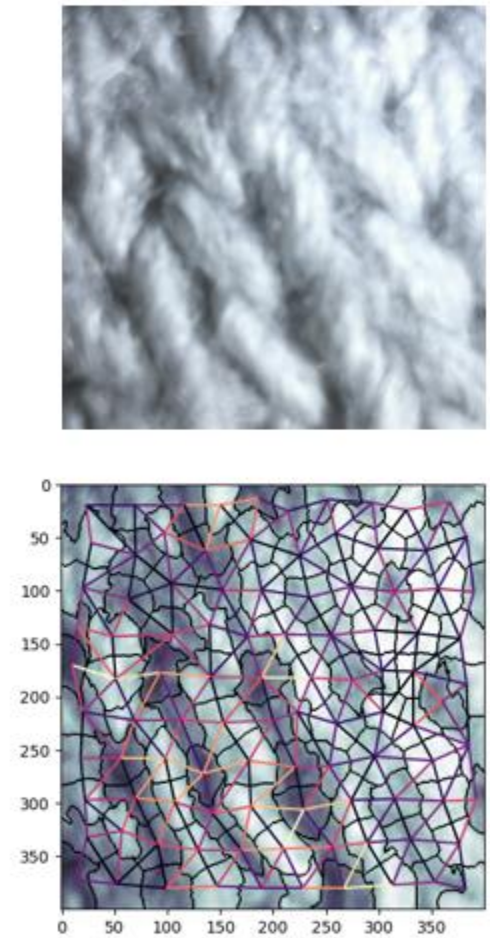

Figure 2 RAG for a microscopic fabric image

When converting RAGs from the original network space to the embedding space, different models can be adopted to include different types of information or to address different goals. Commonly used models include matrix factorization, random walk, deep neural networks and their variations. In this study, DeepWalk, Node2vec and GraRep methods will be used for Euclidean embeddings of RAGs.

DeepWalk (Perozzi, 2014) is a proposed deep learning approach to learn representations of vertices in a RAG that can preserve the neighborhood structure. DeepWalk was developed by observing that the distribution of nodes appearing in short random walks is similar to the distribution of words in natural language. As a result of this observation, the Skip-Gram model (Mikolov et al., 2013), a widely used word representation learning model, is used by DeepWalk to learn the representations of vertices. DeepWalk uses a truncated random walk across a RAG to generate a walking sequence. For each sequence $s = \{v_1, v_2, \dots, v_n\}$

$$\max_{\varphi} P(\{v_{i-w}, \dots, v_{i+w}\} \setminus v_i \mid \varphi(v_i)) = \prod_{j=i-w}^{i+w} P\big(v_j \mid \varphi(v_i)\big).$$

The probability of the neighborhood of v_i and v_i to be in this walk is maximized. In this maximization, w is the window size and $\varphi(v_i)$ is the current representation of the vertex. Finally, effective embedding is performed using hierarchical soft-max (Mikolov et al., 2013).

Node2vec (Grover and Leskovec, 2016) is a method developed after DeepWalk was not descriptive enough to capture the diversity of connection patterns in a RAG. Node2vec defines a flexible concept of RAG neighborhood of a vertex and designs a quadratic random walk strategy to sample neighboring vertices. It can then seamlessly interpolate between the largest initial sampling and the depth initial sampling. Node2vec can closely learn representations that embed vertices with the same mesh community, and can learn from representations where vertices that share similar roles have similar embeddings.

LINE (Tang et al., 2015) has been proposed for large-scale network embeddings and can protect first- and second-order relatives. First-order proximity is the binary proximity observed between vertices. Second-order proximity is determined by the neighborhood similarity of the two vertices. The conditional distribution means that vertices with similar distributions over contexts are similar to each other. By minimizing the KL-divergence of the two distributions and the empirical distributions, respectively, embedding representations of vertices that can maintain first- and second-order closenesses can be obtained. Considering that LINE preserves only first-order and second-order affinities, GraRep (Cao et al., 2015) shows that k-step (k>2) proximity should also be captured when generating global representations of the vertices. Given an adjacency matrix A, the elements of the A^k k-step probability transition matrix represent the probability of a k-step $P(j|i)$ transition from vertex i to vertex j. Similar to the Skip-Gram (Mikolov et al., 2013) model, the loss function of the vertex i can be defined as follows:

$$L_k(i) = \sum_j p(j|i) \, log \, \sigma\left(u_j^T u_i\right) + \lambda \mathbb{E}_{j' \sim p_k(V)}\left[log \, \sigma(-u_i^T u_{j'})\right].$$

In this definition, $\sigma(x) = (1 + e^{-x})^{-1}$, $p_k(V)$ is the distribution over the vertices of the RAG and j is the peak obtained with negative sampling. By reformulating the GraRep loss function as a matrix factorization problem, the singular value decomposition for each k-step loss function can be used directly to extract representations of the vertices. By concentrating on the representations learned from each function, global representations can be obtained.

Convolutional artificial neural network (CNN) is one of the most important architectures of deep learning, which is a multi-layer feed-forward neural network model (Lauzon, 2012). CNN extracts feature from data according to spatial principle. Two-pixel values in related images increase the classification performance of CNN on images. Although CNN architecture is mostly used in image analysis problems, it has recently been used for Euclidean embedding classification problems (Chan et al., 2015; Minaee et al., 2021; Sadad et al., 2021).

CNN architecture basically consists of three layers: convolution, pooling and fully connected layer. Unlike traditional neural networks, the convolution layer performs automatic feature extraction and uses the pool layer for feature reduction. In Figure 3, the general structure of the CNN architecture to be used in this study is shown. Various filters are used in the convolution layer to extract the feature from the input 2D data emerging from the embedding of a RAG. Intermediate processing is applied in the convolution layer and the pooling layers. Thus, the features obtained with the help of the Rectified Linear Unit (ReLU) activation function are not linear. With this feature in the pooling layer, the size of their mapping is reduced. As a result, the computational effort in the next layers is reduced and features in the text are obtained more effectively. The last layer of the convolutional neural network is in the form of a classical fully connected neural network. In this layer, a fully connected structure between artificial neurons represents the properties of the 2D Euclidean data and the target class labels. The new text serves as an input to CNN. When training is complete, CNN gives the estimated class probability.

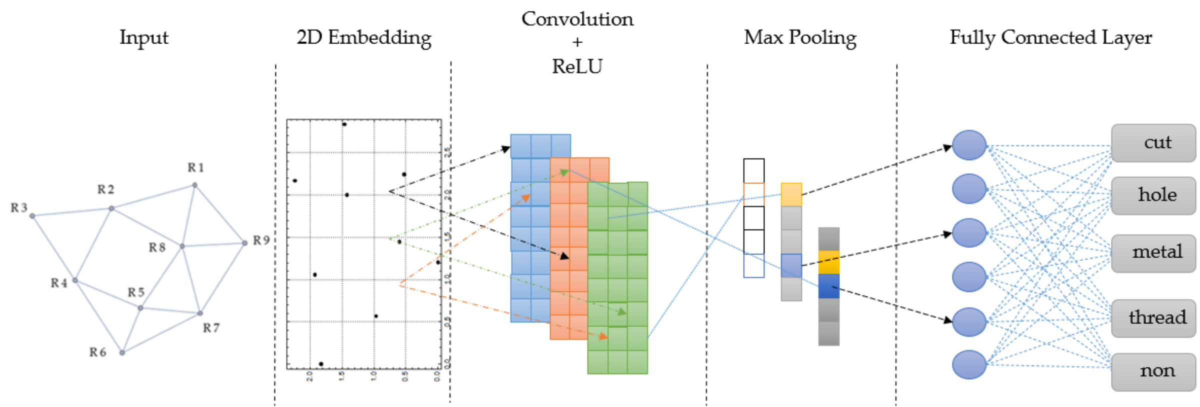

Figure 3 The CNN structure

3. Results

In this study, we use TILDA data set for classification of textile defects. Microscopic fabric data are evaluated in 5 classes as cut, hole, metal contamination, thread defect and defect free. There are 2250 pieces of 64 × 64 image data belonging to each class. The inputs of the classification module are RAG embeddings of image data.

In Figures 4-8, samples of data set are presented. Examples of saliently detect regions and their RAGs are also presented in Figure 9. In order to obtain same volumed embeddings, the number of segmentations is fixed to 25 for all data elements.

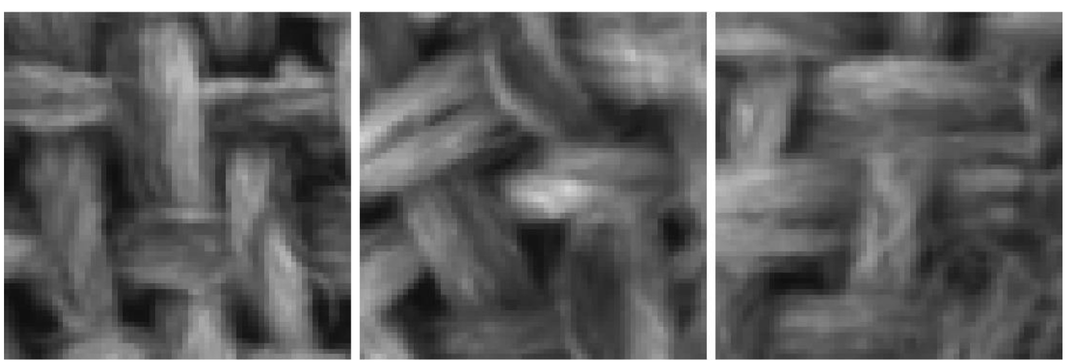

Figure 4 Samples of images labelled by "cut"

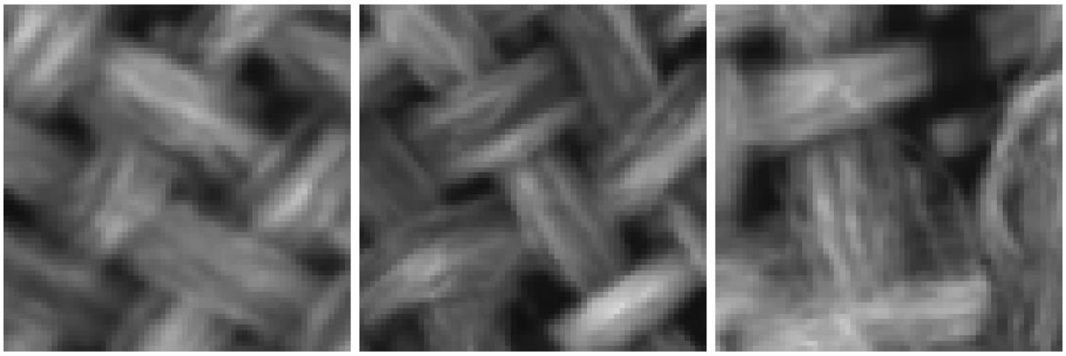

Figure 5 Samples of images labelled by “hole”

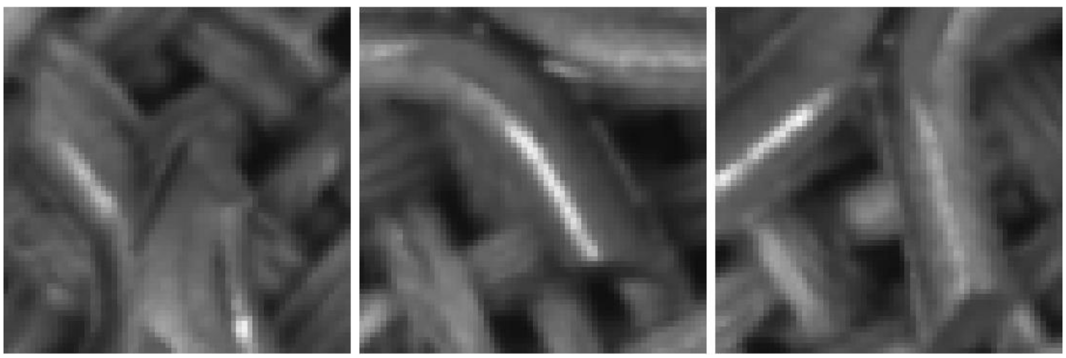

Figure 6 Samples of images labelled by “metal”

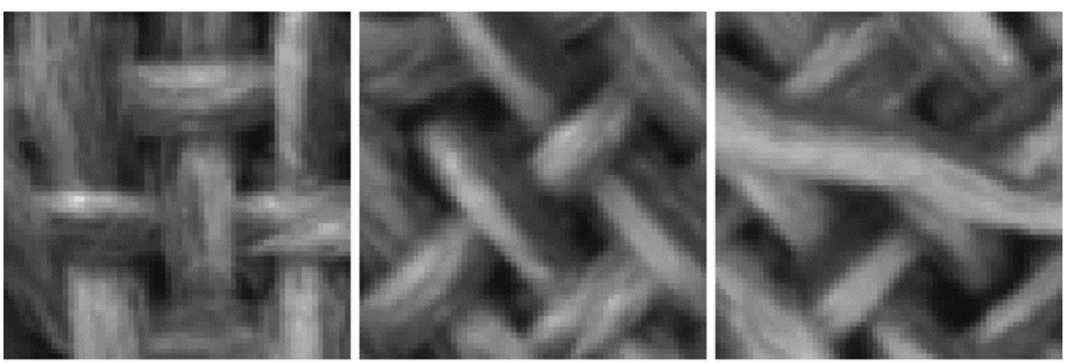

Figure 7 Samples of images labelled by “thread”

Figure 8 Samples of images labelled by "non defect"

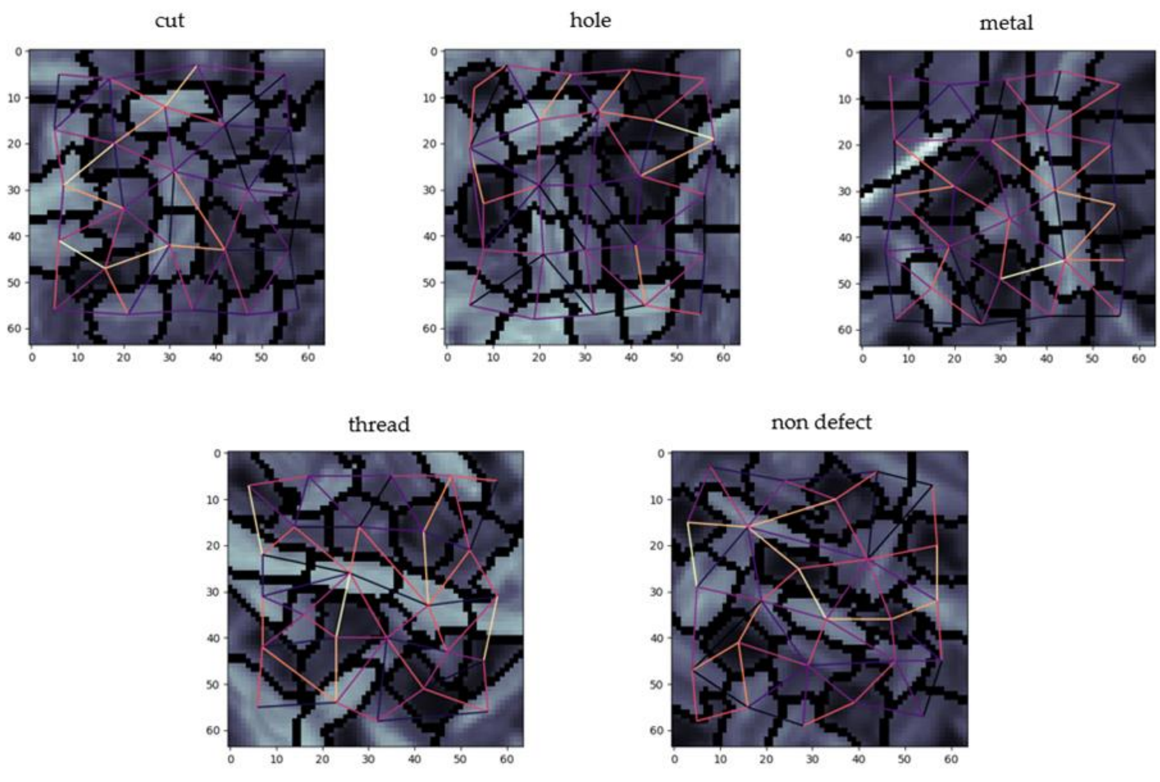

Figure 9 RAG examples

Since we experimentally fixed vertex number of each RAGs to 25, the 2D Euclidean embeddings of RAGs have 25 points for both embedding methods. For each class 1800 images are used for learning and 450 images are used for testing. In Figures 10 – 12, the confusion matrices for each embedding method are presented.

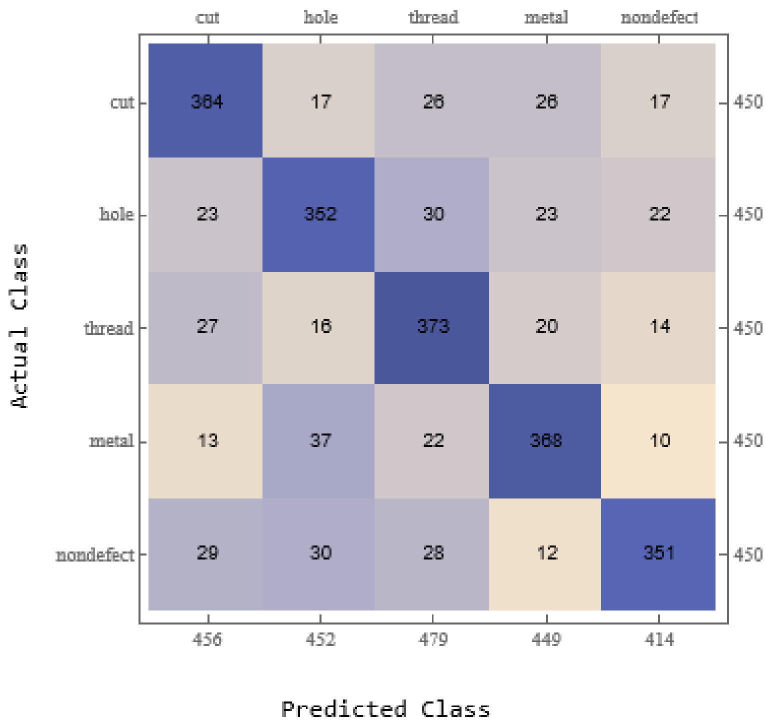

Figure 10 Confusion matrix for DeepWalk Embedding

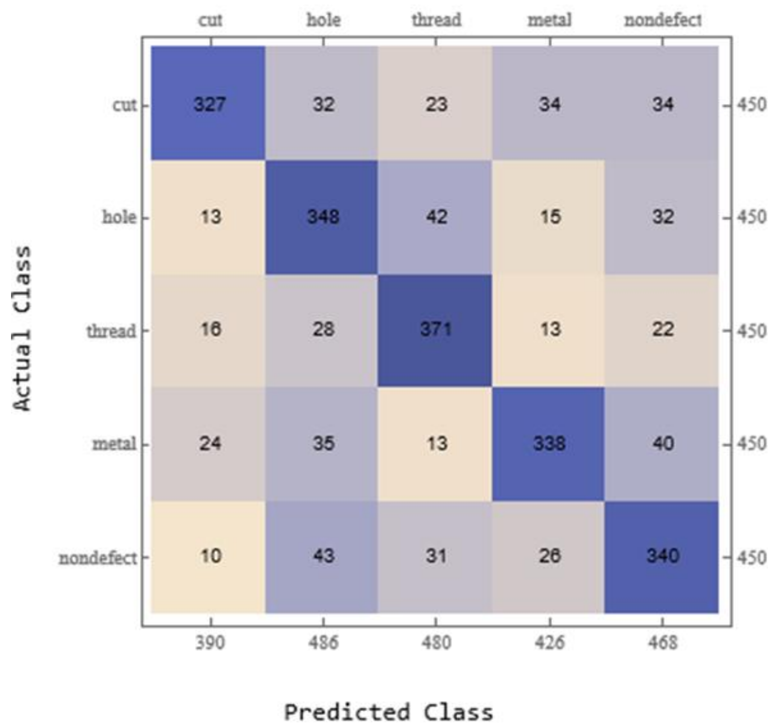

Figure 11 Confusion matrix for Node2Vec Embedding

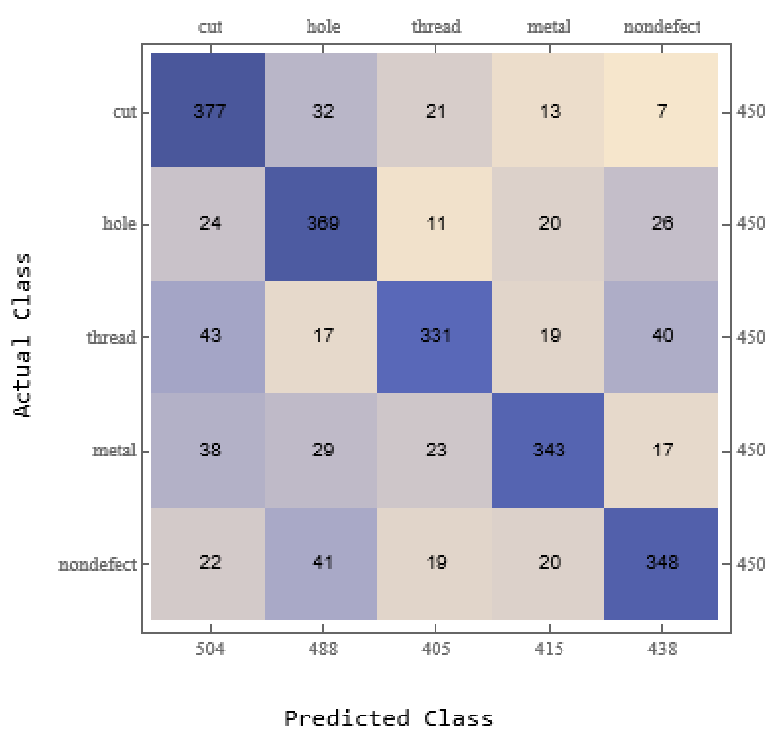

Figure 12 Confusion matrix for GraRep Embedding

4. Discussion and Conclusion

All textile industries compete in fabric production. The course of this competition also depends on the losses caused by faulty fabrics. For this reason, it is very important to detect the fault in the production process of fabric production in the textile industry. In this study, a method is presented for the automated detection of the main defect classes encountered in textile fabric production.

For further analysis of segmented images, it is efficient to develop a rapid mathematical method using the topological properties of the image in each image segment. With this point of view, graph structures are obtained from the neighborhood relations between segments of fabric images in this study.

In this study, a 5-class classification method, including 4 different fabric defect types and defect-free status, is carried out with the deep learning technique. The classification model performed using the classical CNN structure took as input the Euclidean embeddings of the RAGs of the fabric defect classes. There are different techniques in the literature for embedding a combinatorial graph in Euclidean spaces with different dimensions. In order to fully automate the classification process, 3 machine learning-based embedding methods are examined in this study.

When the RAG embeddings obtained by the 3 different methods used in this study are examined, it is seen that the accuracy rates for all methods are above 80%. Among the methods, it is seen from the confusion matrices that the most effective one is DeepWalk. With this approach we present in this study it is possible to establish classification models with more effective results with different and more detailed textile fabric image data sets. In addition, in future studies, graph convolution neural network can be used directly instead of RAG embeddings.

References

[1] Babina, T., Fedyk, A., He, A. X., & Hodson, J. (2020). Artificial intelligence, firm growth, and industry concentration. Firm Growth, and Industry Concentration (November 22, 2020).

[2] Bolton, C., Machová, V., Kovacova, M., & Valaskova, K. (2018). The power of human–machine collaboration: Artificial intelligence, business automation, and the smart economy. Economics, Management, and Financial Markets, 13(4), 51-56.

[3] Thoben, K. D., Wiesner, S., & Wuest, T. (2017). "Industrie 4.0" and smart manufacturing-a review of research issues and application examples. International journal of automation technology, 11(1), 4-16.

[4] Bullon, J., González Arrieta, A., Hernández Encinas, A., & Queiruga Dios, A. (2017). Manufacturing processes in the textile industry. Expert Systems for fabrics production.

[5] Perez, J. J. B., Arrieta, A. G., Encinas, A. H., & Dios, M. A. Q. (2017). Manufacturing processes in the textile industry. Expert Systems for fabrics production. Adcaij-Advances in Distributed Computing and Artificial Intelligence Journal, 6(4), 15-23.

[6] Hanbay, K., Talu, M. F., & Özgüven, Ö. F. (2016). Fabric defect detection systems and methods—A systematic literature review. Optik, 127(24), 11960-11973.

[7] Mahajan, P. M., Kolhe, S. R., & Patil, P. M. (2009). A review of automatic fabric defect detection techniques. Advances in Computational Research, 1(2), 18-29.

[8] Song, L., Li, R., & Chen, S. (2020). Fabric defect detection based on membership degree of regions. IEEE Access, 8, 48752-48760.

[9] Perozzi, B., Al-Rfou, R., & Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 701-710).

[10] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26.

[11] Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 855-864).

[12] Cao, S., Lu, W., & Xu, Q. (2015, October). Grarep: Learning graph representations with global structural information. In Proceedings of the 24th ACM international on conference on information and knowledge management (pp. 891-900).

[13] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015, May). Line: Large-scale information network embedding. In Proceedings of the 24th international conference on world wide web (pp. 1067-1077).

[14] Lauzon, F. Q. (2012, July). An introduction to deep learning. In 2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA) (pp. 1438-1439). IEEE.

[15] Chan, T. H., Jia, K., Gao, S., Lu, J., Zeng, Z., & Ma, Y. (2015). PCANet: A simple deep learning baseline for image classification?. IEEE transactions on image processing, 24(12), 5017-5032.

[16] Sadad, T., Khan, A. R., Hussain, A., Tariq, U., Fati, S. M., Bahaj, S. A., & Munir, A. (2021). Internet of medical things embedding deep learning with data augmentation for mammogram density classification. Microscopy Research and Technique, 84(9), 2186-2194.

[17] Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning--based text classification: a comprehensive review. ACM Computing Surveys (CSUR), 54(3), 1-40.References should be cited in the text by name and year in parentheses.
Example:

[18] Khare, S. K., & Bajaj, V. (2020). Time–frequency representation and convolutional neural network-based emotion recognition. IEEE transactions on neural networks and learning systems, 32(7), 2901-2909.

[19] Oralhan, Z., Oralhan, B., & Yiğit, Y. (2017). Smart city application: Internet of things (IoT) technologies based smart waste collection using data mining approach and ant colony optimization. Internet Things, 14(4), 5.