Fracture mode classification by texture analysis of fracture surface scanning electron microscope images

Akihiro Endo, Yoshiyuki Furuya, Kenji Nagata, Hideki Yoshikawa and Hayaru Shouno

Department of Informatics, Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofu, Japan; Research Center for Structural Materials, National Institute for Materials Science, Tsukuba, Japan; Research and Services Division of Materials Data and Integrated System, National Institute for Materials Science, Tsukuba, Japan

ABSTRACT
Fractography is a practical method of determining the cause of a mechanical-structure failure. Accurate decisions regarding fracture-mode classification require experience and knowledge, which may be difficult to share. Therefore, a database of fracture-surface images should be created, and the decision algorithm typically used by experts must be digitized. In recent years, although image classification using deep learning has been successful, it requires a large amount of data and is difficult to interpret. We propose a step-by-step fracture-mode classification method using fracture-surface images, from low to high magnification, based on the fractography knowledge of experts. Fracture-mode classification is performed using texture features for each patch image that is cut out from the fracture-surface image. The fracture mode for the fracture-surface image is voted based on the results of the patch-image classification. In the classification experiments of three fracture modes, the proposed method classifies the fracture mode in patch images with an accuracy of approximately 90%. Moreover, the classification results of the patch images are voted to correctly classify all fracture-surface images as their respective mode, even from a small dataset.

1. Introduction
Fractography is a practical method of determining the cause of a mechanical-structure failure. When a metallic material fractures, the fracture-surface exhibits features that correspond to a fracture-mode. Fractography is an inverse estimation of the fracture mode by using fracture-surface features. For example, dimples can be observed on a fracture surface when a ductile fracture occurs owing to overload, whereas, in the case of a fatigue fracture, the fracture surface exhibits striations. However, there are many cases in which no striations are observed, even on fatigue-fracture surfaces. Such fracture surfaces are referred to here as macrostructure-dependent (MD) fracture surfaces. In the case of a brittle fracture, features such as cleavage are observed. Additionally, inter-granular cracking and quasi cleavage are observed in the case of hydrogen embrittlement.

Identifying the fracture mode from fracture-surface features requires experience and knowledge. This is because the same mode features can be observed differently depending on the material and measurement conditions. Therefore, a specialist with experience pertaining to fracture surfaces can classify the mode more precisely than a novice. However, sharing the experience and knowledge of experts in detail is difficult. Therefore, each time an expert in the field retires, valuable experience and knowledge are lost. To prevent this loss of information, a database of fracture-surface images must be created, and the mode-classifying algorithm used by experts should be digitized.
Recently, data analysis for fractography has been actively performed [4]. Yamagiwa et al. classified the fracture mode using a deep learning model [5]. They used a deep convolutional neural network to classify the fracture mode from fracture-surface images. By training on approximately 2,700 fracture-surface images, they successfully achieved an accuracy of approximately 92% on the test data.

Deep learning models exhibit good results; however, their decisions are difficult to understand. Therefore, the use of such models is insufficient for the digitization of knowledge and technology, and the reliability of the decision results cannot be verified. Thus, the relationship between the data and fracture modes must be modeled at a level that is easily understood. In addition, deep learning models require a large amount of data, computing resources, and training time. However, large datasets of publicly available fracture-surface images do not exist yet, and datasets focused on specific experiments or materials in laboratories may be considerably small for deep learning models. If the dataset is too small, the deep learning model may not be able to train effective features for fracture-mode classification.

In this study, using fracture-surface images, we propose a fracture-mode classification approach that incorporates the process used by experts in the field of fractography and works well with small datasets. Specifically, some fracture modes are classified based on the measurement magnification. A specific fracture mode is classified from low to high magnification. Fracture-mode classification is performed for each patch image that is cut out from the fracture-surface image using texture features, and the fracture mode of the fracture-surface image is decided by a vote on the results of the patch-image classification. Previous research has shown that texture is an effective representation of fracture-mode features [6]. Therefore, we assume that texture can be used for the classification of fracture modes. Unlike the existing methods using deep learning, the proposed approach clarifies the decision processes and could be applied to small datasets. Here, we describe the results of our attempt to classify dimple, striation, and MD fracture surfaces.

2. Classification of three fracture-surface modes

2.1. Dataset

The specimens used for the experiment were austenitic stainless steel SUS304 and carbon steel SS400, both of which were commercially available plates. The material tests that were conducted include the tensile test, Charpy impact test, fatigue test: SHIMADZU, Japan, and fracture surface observation. For the fatigue test, a fatigue-life test using round bar specimens and a crack-propagation test using compact tension specimens were conducted. Fracture-surface observation was performed by field-emission scanning electron microscopy: JEOL, Japan. Two levels of magnification, x 500 and x 2000, were used for observation. Figure 1 shows examples of three fracture modes. In Figure 1, the left, middle, and right images represent dimple, MD, and striation surfaces, respectively.

2.2. Proposal method

In this study, we attempt to classify these three modes of fracture by using a two-step binary classification. Figure 2 represents the mode-classification procedure used by experts. At first, experts observe the fracture-surface image obtained by a low magnification measurement and check the dimple to determine if it is a ductile or fatigue fracture. Subsequently, in fatigue fractures, the presence or absence of striations is examined from the fracture-surface image obtained by a higher magnification measurement. If striations are observed, it is strong evidence of fatigue fracture; however, if no striations are observed, an MD fracture surface is implied. This procedure is used to classify fracture structures into dimples, striations, and MD fracture surfaces. Therefore, based on this procedure, we propose a hierarchical classification approach from low-magnification to high-magnification measurement images. Specifically, the first step is to perform the binary classification of dimple or not-dimple using the low-magnification measurement images of the fracture surface. In the next step, the remaining fracture images that were classified as not-dimple are used.

Figure 1. Examples of fracture modes.
for the binary classification of striation or MD fracture surfaces using the fracture-surface images measured at a higher magnification. This two-step binary classification, from low to high magnification, enables the classification of three modes of the fracture surface.

Figure 3 shows the flow of the binary fracture-mode classification. We consider the classification of fracture modes for patch images cut from fracture-surface images in each magnification. Texture features are extracted from the patch images, and a binary classification by logistic regression is performed using these features. All patches are classified, and finally, the fracture mode of the image is identified by voting using a set of patches that are cut from the original image.

2.3. Preprocessing

To classify the fracture mode, preprocessing was performed before feature extraction. Figure 4 shows an example of a patch image from preprocessing. In this study, we decided to split a 200 × 200 pixels patch image from a fracture-surface image. The contrast of each patch image may be different, depending on the measurement conditions, even for the same fracture mode and fracture-surface image. If there is such a difference, the same fracture mode may be represented as a different mode in the extracted feature space, and it may not be identified correctly. Therefore, we considered transforming the image to a gradient image, which represents texture in the image, to eliminate local differences in contrast. Laplacian filter[7] was applied to the patch image to convert it to a gradient image. Laplacian filter is a matrix as follows:

\[
\begin{pmatrix}
0 & 1 & 0 \\
1 & -4 & 1 \\
0 & 1 & 0
\end{pmatrix}
\] (1)

After convolution was performed by employing a Laplacian filter, to adjust the format of the image, the pixel values were converted to an absolute value and divided by 4. In this process, decimal points were rounded down. The patch image was large enough to cover the fracture surface, but too computationally time consuming to extract the texture features. Therefore, we performed image downsampling to 100 × 100 pixels for each gradient image and extracted the texture features.

2.4. Texture features

Next, we describe the texture features that are used as features in this study. The difference in the fracture mode appears as a pattern in the fracture-surface image. Such a pattern in the image is considered to be texture. Texture refers to a sequence of pixel values, such as a pattern in an image, and texture analysis is
a method used to characterize such texture[8]. In this study, five basic texture analyses[9] were performed on the patch images, and the statistics calculated from the analysis results were used as the features to classify the fracture mode. The gray-level histogram is an analysis tool that examines the distribution of pixel values in an input image. The gray-level-difference statistic is an analysis tool that examines the difference between the pixel of interest and its neighbors. Like a gray-level histogram, it calculates the distribution of the difference. A co-occurrence matrix is an analysis tool that examines the pixel values that appear in neighboring pixels when a specific pixel value appears in the observed pixel. A run-length matrix examines a sequence of specific pixel values, which is called a ‘run,’ and its length is the run length. This analysis computes the frequency of occurrence of the various run lengths for each pixel value. The Fourier power spectrum is a texture-analysis tool that examines the representation of an image in spatial-frequency space. Table 1 shows number of features of each analysis.

Table 1. List of texture analysis.

| Analysis                     | Number of features |
|------------------------------|--------------------|
| Graylevel histogram          | 7                  |
| Graylevel difference         | 7                  |
| Co-occurrence matrix         | 6                  |
| Run length matrix            | 5                  |
| Fourier power spectrum       | 14                 |
| Total                        | 39                 |
From these texture analyses, we computed a total of 39 statistics from the patch images. Detailed definitions are given in the Appendix A.

### 2.5. Logistic regression

In this study, the extracted features were used as input vectors to classify the fracture mode. We performed binary classification via linear logistic regression[10], which is a model that outputs the probability of the predicted label through a weighted linear combination of features. When the input of a $p$-dimensional vector is $x_i = (x_{i1}, \ldots, x_{ip})$ and its label is $y_i \in \{0, 1\}$, the probability is represented as follows:

$$P(y_i = 1|x_i; \beta, \beta_0) = \frac{1}{1 + e^{-(\beta_0 + \beta^T x_i)}}. \quad (2)$$

In this case, $\beta = (\beta_1, \ldots, \beta_p)$ is the weight vector, and $\beta_0$ is the bias term. If $P(y_i = 1|x_i)$ is close to 1, $x_i$ is likely to be label 1, and if it is close to 0, $x_i$ is likely to be label 0. The weights, $\beta$, and bias terms, $\beta_0$, were determined to minimize the following objective function using $N$ data:

$$L(\beta, \beta_0) = -\log \sum_i^N P(y_i|x_i; \beta, \beta_0), \quad (3)$$

$$= - \sum_i^N (1 - y_i) \log(1 - P(y_i = 1|x_i; \beta, \beta_0)) + y_i \log P(y_i = 1|x_i; \beta, \beta_0). \quad (4)$$

### 2.6. Classification by voting

We considered voting for the classification of the patch-image classification results for the decision on the fracture mode of the fracture surface image. First, $D$ patch images cut from a single fracture-surface image are classified into a fracture mode using a trained classifier. From the probability of each patch image, which is the output of the classifier, the average probability is calculated and treated as the probability of the fracture mode of a single fracture-surface image:

$$\text{Voting}(X; \beta, \beta_0) = \frac{1}{D} \sum_x P(y_i = 1|x_i; \beta, \beta_0). \quad (5)$$

$X$ is a set of feature vectors of $D$ patch images cut from a single fracture-surface image. For classification, a threshold with a probability of .5 is used to assign the fracture mode.

### 3. Experiments

The proposed classification approach was used to classify the fracture-surface images into three fracture modes: dimple, striation, and MD fracture surface. To demonstrate the effectiveness of the proposed approach, we performed two classification experiments based on Figure 3. In the first experiment, we attempted to classify the patch images into fracture modes and confirmed the effectiveness of the texture features and patch images. In the second experiment, we classified the fracture-surface images to the fracture modes by voting the results of the patch classification. In each classification experiment, we assumed two situations: low magnification and high magnification (Figure 2). In a low-magnification situation, images acquired at $x$ 500 were classified into dimple or not-dimple. In total, 13 dimple, 10 striation, and 19 MD fracture-surface images were used. However, in a high-magnification situation, we used 24 striation and 10 MD fracture-surface images acquired at $x$ 2000, and these images were classified. Additionally, 35 patch images were cut from each fracture-surface image, and two experiments were performed. Figure 5 shows examples of the patch images used in the experiment.

To evaluate each experiment, the dataset was cross-validated[11]. Cross-validation splits the dataset into training and validation datasets and uses them to train and evaluate the model. In particular, in K-fold cross-validation, the dataset is divided into $K$ sets.

![Figure 5. Examples of patch images.](image-url)
Additionally, the model is evaluated by performing the procedure K times, using one of the K sets for validation and the remaining sets to train the model. However, when splitting the dataset, patch images cut from the same image may be used for both training and validation. In such a case, we considered that the validation data is not appropriate for evaluation. Therefore, in this study, we evaluated the proposed approach using leave-one-image-out cross-validation, in which training is performed on all except for one fracture-surface image, and validation is performed on the image that is not used for training. For example, in a low-magnification situation, one image is used for validation, and the remaining 41 images are used for training. Similarly, in a high-magnification situation, one image is used for verification, and the remaining 33 images are used for training. This procedure was performed for each image, and the model was evaluated.

### 3.1. Classification of patch images

The fracture-mode classification of patch images was performed using texture features. Because logistic regression is a binary classification model, it follows the decision flow of an expert (see Figure 2) and decides between dimple or not-dimple in a low-magnification situation, and between striation or not-striation in a high-magnification situation. Table 2 and Table 3 show the confusion matrices in low- and high-magnification situations, respectively. The confusion matrix summarizes the classification results, which show the output of the logistic regression in the row direction and the true label of the input in the column direction; the diagonal components are the number of patch images that resulted in the correct label during classification. The accuracy of the confusion matrix at low magnification is approximately 90%, and that at high magnification is 87%. Based on this result, we confirm that we could extract sufficient useful information from the patch images. However, when decisions were made based on patch images alone, an error rate of approximately 10% occurred in both low- and high-magnification situations; therefore, it is necessary to devise methods to improve the accuracy.

### 3.2. Classification of fracture-surface images by voting

The next experiment classified the fracture images by voting the patch-image classification results. Figure 6 shows the classification results for each patch image, as shown on the left, for one dimple fracture-surface image. In Figure 6, the color indicates the probability of a dimple, where red and blue represent probabilities that are higher and lower than .5, respectively. Some of the patch images are misclassified; however, most of the patches are labeled correctly. From this result, we assume that the fracture mode of the fracture-surface image can be correctly classified by voting the patch classification results. The results of the classification of fracture-surface images by voting the classification results for each patch are shown in Figure 7. The left subfigure of Figure 7 displays the results obtained in a low-magnification situation. The horizontal axis is the image number, the red bars indicate the dimple image, and the blue bars indicate other fracture modes. When a threshold value of .5 was used to classify the fracture mode, it can be observed that all fracture-surface images were correctly classified as dimple or not-dimple. The right subfigure of Figure 7 shows the results obtained in a high-

| Table 2. Confusion matrix of low magnification. |
|-----------------------------------------------|
|          | Dimple | Striation & MD |
| Dimple   | 368    | 87             |
| Striation & MD | 67    | 948            |

| Table 3. Confusion matrix of high magnification. |
|-----------------------------------------------|
|          | Striation | MD |
| Striation | 620      | 66  |
| MD        | 68       | 282 |

Figure 6. Result of patch images in one fracture image. If the probability of dimple is higher, the color is more red; otherwise, it is more blue.
magnification situation; the red bars indicate striations, and blue bars indicate an MD-fracture surface. Similar to a low-magnification situation, when a threshold value of 0.5 was used to classify the fracture mode, it can be observed that all the fracture-surface images were correctly classified as striation or MD fracture surfaces. From these results, we confirmed that the proposed approach is able to classify three fracture modes for fracture-surface images.

4. Discussion

This section presents a discussion on the results and the proposed approach.

First, we classified fracture modes using texture features extracted from patch images. As indicated by the confusion matrix (Table 2 and Table 3), the accuracy of each patch classification is generally good. If dimple is labeled as 1 in a low-magnification situation, then the precision, recall, and F1-score were above 0.8. Moreover, if striation is labeled as 1 in a high-magnification situation, then precision, recall, and F1-score were above 0.9. From these results, we consider that a pattern in the patch image reasonably represents the specific fracture mode and texture features are effective in classifying the fracture mode. The results of the voting classification (Section 3.2) also suggest that the proposed approach is practical. However, future work will consider an appropriate patch size to achieve a more accurate classification. In this study, a patch size of 200 × 200 pixels was used. This size is considered reasonable considering that relatively accurate classification was achieved.

Next, we discuss the texture features. The classification of fracture modes from the fracture surface was performed using features with classical texture analysis. Owing to the high classification accuracies achieved in this study, we can assume that the texture features treated herein were practical. However, if new fracture modes are added to the classification or new materials are treated in the future, features other than those used in this study may be required. Thus, in future work, the usefulness of the texture features used in this study must be considered and features and image preprocessing should be improved.

Finally, we discuss the classification approach. In this study, we used logistic regression as a classifier. This model outputs the probability value of the predicted label for the input. Using a model that outputs a probability value is more effective at interpreting the results than a model that simply outputs a label. Also, simple models such as logistic regression often work better in situations where the number of data is limited compared to complex models such as deep learning, which generally require a large amount of data. Especially, using deep learning for fracture mode classification, more than 1,000 images were needed for training the model, but using the proposed method, only 42 images were sufficient. In classifying the three fracture modes, we applied the binary classification model and logistic regression, hierarchically based on the magnification. The proposed method follows the actual process of classification of these fracture modes used by experts. In the experimental results, the proposed method could classify three types of fracture forms with sufficiently high accuracy. Therefore, the proposed method successfully replicates the classification process of experts. Based on these points of view, we consider that the proposed method would be well-suited for the digitization of knowledge.

5. Conclusion

In this study, we attempted to classify fracture modes from fracture-surface images. Texture analysis was applied to the fracture surface images, and the extracted features enabled the classification of the fracture modes. Based on the classification experiments, we confirmed that the fracture mode of a fracture-surface image can be determined by voting on the classification results of patch images; however, some classification errors were noted. Therefore, we considered that the proposed approach

![Figure 7. Probability of the image being classified as dimple, striation, or MD. Left shows the probability of dimple. Right shows the probability of striation.](image-url)
would work for fracture mode classification even with small datasets. In future work, whether the proposed approach can be extended to the classification of other modes, such as cleavage and intergranular cracking, should be investigated. We also intend to investigate the optimal patch size for the classification of fracture modes. As a result, this method would be improved to assist with the failure analysis of real accidents.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Yoshiyuki Furuya https://orcid.org/0000-0002-3039-5280
Kenji Nagata https://orcid.org/0000-0001-9894-4461
Hayaru Shouno https://orcid.org/0000-0002-2412-0184

References

[1] Wiley J, Sons L. Failure analysis. In: Metal failures; 2013. p. 1–26. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118671023.ch1
[2] Whiteson BV, Phillips A, Kerlins V. Electron fractography handbook; 1965. Available from: https://www.osti.gov/biblio/4605301
[3] Jacoby G. Fractographic methods in fatigue research. Exp Mech. 1965;5(3):1741–2765. DOI:10.1007/BF02323211.
[4] Adachi Y, Taguchi M, Hirokawa S. Microstructure recognition by deep learning. Tetsu-to-Hagane. 2016;102(12):722–729; advpub. DOI:10.2355/tetsuto-hagane.TETSU-2016-035.
[5] Yamagiwa K. Application of deep-learning for classification of fracture surface’s sem image. J Soc Mater SciJap. 2020;69(9):644–649.
[6] Naik D, Kiran R. Identification and characterization of fracture in metals using machine learning based texture recognition algorithms. Eng Fract Mech. 2019;219:106618. DOI:10.1016/j.engfractech.2019.106618.
[7] Moulin P. Multiscale image decompositions and wavelets. In: Bovik A, editor. Handbook of image and video processing (second edition). 2nd ed. Burlington: Academic Press; 2005. p. 347–359. Communications, Networking and Multimedia. Available from: https://www.sciencedirect.com/science/article/pii/B978012119792650084X
[8] Tuceryan M, Jain AK. Texture analysis. In: Handbook of pattern recognition and computer vision; 1993. p. 235–276. Available from: https://www.worldscientific.com/doi/abs/10.1142/9789814343138_0010
[9] Sugata Y, Kido S, Shouno H. Comparison of two-dimensional with three-dimensional analyses for diffuse lung diseases from thoracic CT images. Med Imaging Inf Sci. 2008;25(3):43–47.
[10] Bishop CM. Pattern recognition and machine learning. Springer; 2006.
[11] Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2; San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.; 1995. p. 1137–1143; IJCAI’95.
Appendix A. Texture Analysis

To examine the properties of the texture, texture analysis is applied to the image. In this section, five typical texture analyses are described. We let the number of pixel values in the target image be \( L \) and the pixel value be \( l (0 \leq l \leq L - 1) \).

A.1. Gray-level histogram

Graylevel histogram presents the frequency of \( l \). We let the probability of \( l \) be \( P(l) \), and from \( P(l) \), we obtain the following seven features.

\[
\text{mean} (\mu) = \sum_{l=0}^{L-1} lP(l) \quad (A1)
\]

\[
\text{variance} (\sigma) = \sum_{l=0}^{L-1} (l - \mu)^2P(l) \quad (A2)
\]

\[
\text{contrast} = \sum_{l=0}^{L-1} l^2P(l) \quad (A3)
\]

\[
\text{skewness} = \frac{1}{\sigma^3} \sum_{l=0}^{L-1} (l - \mu)^3P(l) \quad (A4)
\]

\[
\text{kurtosis} = \frac{1}{\sigma^4} \sum_{l=0}^{L-1} (l - \mu)^4P(l) \quad (A5)
\]

\[
\text{energy} = \sum_{l=0}^{L-1} P^2(l) \quad (A6)
\]

\[
\text{entropy} = -\sum_{l=0}^{L-1} P(l) \log P(l) \quad (A7)
\]

These features are basic statistics, \( \mu \) and \( \sigma \) are the average and the variance of the pixel value in the target image, respectively. If the histogram is biased toward higher, contrast shows higher. If the histogram is asymmetric, skewness becomes higher. Kurtosis shows how much the histogram is concentrated around \( \mu \). The larger a particular \( P(l) \) is, the larger the energy; conversely, entropy takes on a larger value the larger the various \( P(l) \) are.

A.2. Graylevel difference

In the target image, we consider pixel pair \((i,j)\). Pixel \(j\) is in a position to pixel \(i\) that is direction \( \theta \) and range \( r \). Pixel values of pixel \(i\) and \(j\) let \(l_i\) and \(l_j\), and difference \(l'\) are defined as follows:

\[
l' = |l_i - l_j| \quad (A8)
\]

We defined probability \( P_d(l') \) as difference statistics, where \( \delta = (\theta, r) \). Follow formula A1 to A7, \( P(l) \) is replaced to \( P_d(l') \), and seven features of \( \delta \) are calculated. Features are calculated for each \( \delta \), and we let their averages with respect to \( \delta \) be seven features of Graylevel difference. In this study, we considered the nearest eight pixels with \( r = 1 \) and calculated seven features about four directions with \( \theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ \).

A.3. Co-occurrence Matrix

As in A.2, we consider pixel pair \((i,j)\) and \(\delta\). The co-occurrence Matrix is based on \( P_d(l_i, l_j) \), which is the probability that the pixel values of \(i\) and \(j\) are \(l_i\) and \(l_j\). From this \( P_d(l_i, l_j) \), we obtain the following six features.

\[
\text{energy} = \sum_{l=0}^{L-1} \sum_{l=0}^{L-1} [P_d(l_i, l_j)]^2 \quad (A9)
\]

\[
\text{contrast} = \sum_{k=0}^{L-1} k^2 P_{x-y}(k) \quad (A10)
\]

\[
\frac{\sum_{l=0}^{L-1} \sum_{l=0}^{L-1} l l' P_d(l_i, l_j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (A11)
\]

\[
\text{variance} = \sum_{l=0}^{L-1} \sum_{l=0}^{L-1} (l - \mu_x)^2 P_d(l_i, l_j) \quad (A12)
\]

\[
\text{IDM} = \sum_{l=0}^{L-1} \sum_{l=0}^{L-1} \frac{1}{1 + (l - l_j)^2} P_d(l_i, l_j) \quad (A13)
\]

\[
\text{entropy} = -\sum_{l=0}^{L-1} \sum_{l=0}^{L-1} P_d(l_i, l_j) \log \{P_d(l_i, l_j)\} \quad (A14)
\]

In the above formulas, \( P_x(l_i), P_y(l_j), P_{x-y}(k), \mu_x, \mu_y, \sigma_x, \sigma_y \) and \( k \) are calculated according to the following formulas:

\[
P_x(l_i) = \sum_{l=0}^{L-1} P_x(l_i, l_j) \quad (A15)
\]

\[
P_y(l_j) = \sum_{l=0}^{L-1} P_y(l_i, l_j) \quad (A16)
\]

\[
P_{x-y}(k) = \sum_{l=0}^{L-1} \sum_{l=0}^{L-1} P_x(l_i, l_j) \quad (A17)
\]

\[
\mu_x = \sum_{l=0}^{L-1} l P_x(l_i) \quad (A18)
\]

\[
\mu_y = \sum_{l=0}^{L-1} l P_y(l_j) \quad (A19)
\]

\[
\sigma_x^2 = \sum_{l=0}^{L-1} (l - \mu_x)^2 P_x(l_i) \quad (A20)
\]

\[
\sigma_y^2 = \sum_{l=0}^{L-1} (l - \mu_y)^2 P_y(l_j) \quad (A21)
\]

\[
k = |l_i - l_j| \quad (A22)
\]

In the Co-occurrence Matrix and Graylevel difference, we considered the nearest 8 pixels with \( r = 1 \) and calculated 6 features about 4 directions with \( \theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ \).

A.4. Run length matrix

A sequence of pixels with the same pixel value in a certain direction is called a ‘run,’ and its length is called a ‘run length.’ We consider the probability \( P_0(l, d) \), where \( \theta \) is direction, \( l \) is pixel value, and \( d \) is the run length. The run length matrix presents the probabilities of run length for each direction and each pixel value. From the run length matrix, we calculate the following 5 features.
SRE = \( \frac{1}{P_\theta} \sum_{l=0}^{L-1} \sum_{d=1}^{d_{\text{max}}} P_\theta(l, d) \) \( \sum_{l=0}^{L-1} \sum_{d=1}^{d_{\text{max}}} r^2 P_\theta(l, d) \) \( \sum_{l=0}^{L-1} \sum_{d=1}^{d_{\text{max}}} \) \( \sum_{l=0}^{L-1} \sum_{d=1}^{d_{\text{max}}} \) \( P_\theta(l, d) \) \( P_\theta(l, d) \) \( P_\theta(l, d) \)

A is the number of pixels in the target image, and \( r_{\text{max}} \) is the maximum value which \( r \) could acquire. SRE is higher if there are many short runs. On the contrary, LRE is higher if there are many long runs. GLN presents the non-uniformity of \( l \). If a run occurs at various \( l \), GLN becomes higher. RLN presents the non-uniformity of \( d \). RPC is the run percentage in the target image.

### A.5. Fourier power spectrum

To investigate the frequency response of the target image, the discrete Fourier transform is applied to obtain the power spectrum. The target image \( f(x, y) \) is \( M \times N \) pixels, and is converted by the discrete Fourier transform to \( F(x, y) \). The power spectrum \( P(s, t) \) is defined by the following formulas:

\[
F(s, t) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp \left( -2\pi i \frac{sx}{M} + \frac{ty}{N} \right) \]

where, we let \( s = 0, 1, \ldots, M - 1 \) and \( t = 0, 1, \ldots, N - 1 \), and \( i \) is an imaginary unit. \( P(s, t) \) is transformed by the coordinate axes so that the center is \( (u, v) = (0, 0) \), and then performed the polar coordinate transformation represented by the direction \( \theta \) and the distance \( r \), and the result is \( P(\theta, r) \). From \( P(\theta, r) \), we calculate \( P(\theta) \) and \( P(r) \).

\[
P(\theta) = \sum_{\theta - \Delta \theta/2}^{\theta + \Delta \theta/2} \sum_{r = r_{\text{max}} - \Delta r/2}^{r_{\text{max}} + \Delta r/2} P(\theta, r)
\]

\[
P(r) = \sum_{r = r_{\text{max}} - \Delta r/2}^{r_{\text{max}} + \Delta r/2} \sum_{\theta = 0}^{\theta_{\text{max}}} P(\theta, r)
\]

where, \( r_{\text{max}} \) is the maximum of \( r \) and \( \theta_{\text{max}} \) is the maximum of \( \theta \). \( \Delta \theta \) and \( \Delta r \) are the measurement width of \( \theta \) and \( r \) respectively. Figure A1 shows concepts of \( P(\theta) \) and \( P(r) \). In the Fourier space, \( P(\theta, r) \), \( P(\theta) \) is the sum of the fan-shaped region and \( P(r) \) is the sum of the concentric regions. These represent the directionality and roughness of the texture. \( P(\theta) \) and \( P(r) \) are normalized by dividing them by the total number of pixels in the region, and we compute \( \tau \) features each from \( P(\theta) \) and \( P(r) \) as follow formula A1 to A7.

![Figure A1. Concepts of P(θ) (left) and P(r) (right).](image-url)