Classification of Hepatocellular Carcinoma and Liver Abscess by Applying Neural Network to Ultrasound Images

Sendren Sheng-Dong Xu,[1] Chun-Chao Chang,[2,3] Chien-Tien Su,[4,5*]
Pham Quoc Phu,[1] Tifany Inne Halim,[1] and Shun-Feng Su[6]

[1]Graduate Institute of Automation and Control, National Taiwan University of Science and Technology,
Da'an District, Taipei 10607, Taiwan
[2]Division of Gastroenterology and Hepatology, Department of Internal Medicine,
Taipei Medical University Hospital, Xinyi District, Taipei 11031, Taiwan
[3]Division of Gastroenterology and Hepatology, Department of Internal Medicine, School of Medicine,
College of Medicine, Taipei Medical University, Xinyi District, Taipei 11031, Taiwan
[4]School of Public Health, College of Public Health, Taipei Medical University, Xinyi District, Taipei 11031, Taiwan
[5]Department of Family Medicine, Taipei Medical University Hospital, Xinyi District, Taipei 11031, Taiwan
[6]Department of Electrical Engineering, National Taiwan University of Science and Technology,
Da'an District, Taipei 10607, Taiwan

In diagnostic ultrasound, an ultrasound transducer converts an electrical signal into an ultrasound pulse, which enters the tissue from the body surface. At the surface, an echo appears. The probe senses and receives the echo, and all the echoes are converted back to signals and graphics, which can be analyzed by medical staff. We studied the neural network (NN)-based classification of hepatocellular carcinoma (HCC) and liver abscess using texture features of ultrasound images. From 79 cases of liver diseases (44 liver cancer and 35 liver abscess cases), we extracted 52 features of the gray-level co-occurrence matrix (GLCM) and 44 features of the gray-level run-length matrix (GLRLM), giving a total of 96 features. We used three feature selection models to distinguish these two liver diseases: sequential forward selection (SFS), sequential backward selection (SBS), and F-score. We proved that our developed system can be used to classify liver cancer and liver abscess using an NN with an accuracy of 88.375%, which can provide diagnostic assistance for inexperienced clinicians.

1. Introduction

There are many liver diseases, for example, liver cancer (hepatocellular carcinoma, HCC) and liver abscess. Liver cancer has a high mortality rate. Depending on the stage of the disease, the treatment also varies between surgery, radiation therapy, chemotherapy, tumor ablation, embolization therapy, targeted therapy, and many others. Even though liver biopsy is effective for obtaining a correct diagnosis, it may generate side effects in patients such as pain,

infection, or injuries in the subsequent treatment. Because of the various risks and undesired effects, there are many other approaches to help diagnose liver disease. Ultrasound imaging is a feasible approach, and a computer-aided diagnosis (CAD) system can help an inexperienced clinician in diagnostic evaluation.

Medical ultrasound imaging is based on the pulse–echo principle. An ultrasound transducer converts an electrical signal into an ultrasound pulse, which enters the tissue from the body surface. At the surface, an echo appears. The probe senses and receives the echo, and all the echoes are converted back to signals and graphics, which can be seen by medical staff. In medical ultrasound, a coupling gel is used as a universal medium to avoid excessive reflection caused by the tiny amount of air between the probe and the skin.[1,2]

On the other hand, neural networks (NNs) are a powerful technique for solving research problems. For example, a radial basis function NN is used to design a control law for a derived mathematical kinematic model of mobile robots.[3,4] Chien *et al.* applied a multilayer perceptron (MLP) NN to an impulse noise detector for power-line-based sensor networks.[5] Moreover, an MLP NN has been applied to classification for biomedical image processing.[6] Here, we propose NN-based classification for the CAD of images obtained from ultrasound imaging, which has several advantages over liver biopsy such as no radiation risk, low cost, easy operation, and non-invasiveness. We applied the gray-level co-occurrence matrix (GLCM)[7–9] and the gray-level run-length matrix (GLRLM)[7,10] as textural features with three feature selection models: sequential forward selection (SFS),[7,9,11] sequential backward selection (SBS),[7,9,12] and F-score.[7,13]

2.   Feature Extraction

We retrieved the images for analysis from the Medical University Hospital in Taipei, which consisted of 44 cases of HCC and 35 cases of liver abscess: in total, 79 cases of liver disease. For each case, we selected a 32 × 32-pixel region of interest (ROI) inside marked boundaries and converted it to a 256-grayscale BMP file using MATLAB for convenient processing as shown in Fig. 1. We sampled 400 ROIs from the images for each disease, which were used for training and testing. From each of the ROIs, we extracted 96 features (52 GLCMs and 44 GLRLMs). The GLCM[7–9] is represented by a matrix depicting how different combinations of gray levels exist in an image.

The GLCM feature extraction method consists of two steps: (1) co-occurrence matrix calculation and (2) the computation of texture features from the co-occurrence matrix. The GLCM feature extraction results, also called the Haralick features, are extracted from each image and shown in Table 1. The other method employed in this research is the use of the GLRLM[7,9] to compute four matrices for horizontal, vertical, and diagonal directions, i.e., 0, 45, 90, and 135°, to produce a run-length matrix. The results of the calculation are called texture descriptors, and each descriptor is unique for each texture. We extracted the 11 most often used features from the run-length matrices, which are shown in Table 2.
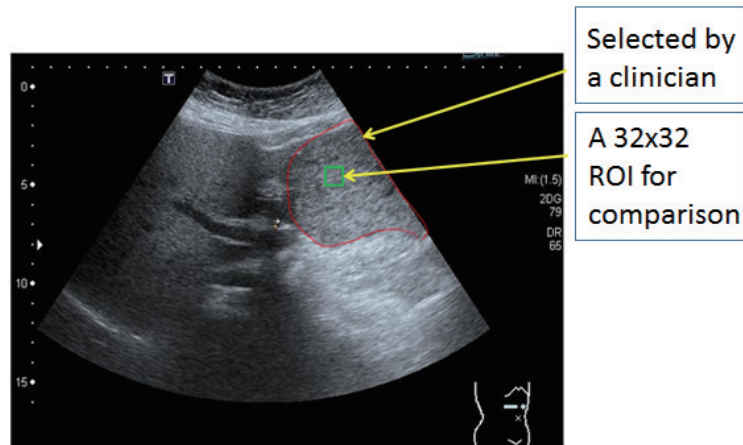
Fig. 1.　　(Color online) Input ultrasound image to be processed.

Table 1
GLCM features.

Code	Feature	Equation	Value
f_1	Energy	$\sum_{i=0}^{l-1}\sum_{j=0}^{l-1}P_{d,\theta}^2(i,j)$	0.1386
f_2	Entropy	$\sum_{i=0}^{l-1}\sum_{j=0}^{l-1}P_{d,\theta}(i,j)\cdot\log P_{d,\theta}(i,j)$	2.0915
f_3	Contrast	$\sum_{i=0}^{l-1}\sum_{j=0}^{l-1}\lvert i-j\rvert^2\cdot P_{d,\theta}(i,j)$	0.9960
f_4	Correlation	$\sum_{i=0}^{l-1}\sum_{j=0}^{l-1}P_{d,\theta}(i,j)\dfrac{(i-\mu_x)(j-\mu_y)}{\sigma_x\sigma_y}$	0.5119
f_5	Homogeneity	$\sum_{i=0}^{l-1}\sum_{j=0}^{l-1}\dfrac{1}{1+(i-j)^2}P_{d,\theta}(i,j)$	0.7213
f_6	Sum average	$\sum_{i=0}^{2(l-1)}i\cdot P_{x+y}(i)$	2.3260
f_7	Sum entropy	$-\sum_{i=0}^{2(l-1)}P_{x+y}(i)\cdot\log P_{x+y}(i)$	1.5155
f_8	Sum variance	$\sum_{i=0}^{2(l-1)}(i-f_7)^2\cdot P_{x+y}(i)$	2.8829
f_9	Difference average	$\sum_{i=0}^{(l-1)}i\cdot P_{x-y}(i)$	0.4980
f_{10}	Difference variance	$\sum_{i=0}^{l-1}(i-f_9)^2\cdot P_{x-y}(i)$	0.5542
f_{11}	Difference entropy	$-\sum_{i=0}^{l-1}P_{x-y}(i)\cdot\log P_{x-y}(i)$	1.0107
f_{12}	Information measures of correlation feature 1	$\dfrac{HXY-HXY1}{\max(HX,HY)}$	−0.3713
f_{13}	Information measures of correlation feature 2	$\left(1-\exp\left[-2(HXY2-HXY)\right]\right)^{\frac{1}{2}}$	0.7840

Table 2
GLRLM features.

Code	Feature	Equation	Value
SRE	Short-run emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{Q(i,j)}{j^2}$	0.8269
LRE	Long-run emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}Q(i,j)\cdot j^2$	1.6923
SRHGE	Short-run high-gray-level emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{Q(i,j)\cdot i^2}{j^2}$	5.9423
LRLGE	Long-run low-gray-level emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{Q(i,j)\cdot j^2}{i^2}$	0.4348
LRHGE	Long-run high-gray-level emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}Q(i,j)\cdot i^2\cdot j^2$	14.3077
LGRE	Low-gray-level run emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{Q(i,j)}{i^2}$	0.3371
HGRE	High-gray-level run emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}Q(i,j)\cdot i^2$	7.6154
SRLGE	Short-run low-gray-level emphasis	$\dfrac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{Q(i,j)}{i^2\cdot j^2}$	0.3126
RLN	Run-length nonuniformity	$\dfrac{1}{n_r}\sum_{j=1}^{N}\left(\sum_{i=1}^{M}Q(i,j)\right)^2$	8.3846
GLN	Gray-level nonuniformity	$\dfrac{1}{n_r}\sum_{i=1}^{M}\left(\sum_{j=1}^{N}Q(i,j)\right)^2$	3.3077
RPC	Run percentage	$\dfrac{n_r}{n_p}$	0.8125

3. Feature Selection

The main idea of our feature selection methods is to keep the useful features while eliminating those that contain little or no predictive information. The advantages of using feature selection are reduced computation and cost, improved accuracy, and greater understanding of the difference between HCC and liver abscess. We used three feature selection methods: SFS, SBS, and F-score.

The SFS method starts with an empty set and adds the next selected feature $x^* = \text{argmax}[J(Y_k + x)]$ with $x \notin Y_k$, in which $J(Y_k + x)$ is the highest objective function. This is repeated continuously until a predefined number of features are selected. SBS works the opposite way: it starts from a full set of features and removes the worst feature continuously until a predefined number of features are left.

The F-score measures discrimination from a given training vector. The higher the F-score, the more it discriminates between the positive and negative sets. The disadvantage of the

F-score is that it cannot reveal shared information between features. We used two F-score methods. The first method, called the search-all method, computed all the features. The second method, called the threshold method, selected four thresholds for each feature and six thresholds for all features (discussed in Sect. 6) where the gap between the low and high F-scores is considerable.

4. NN

Recently, numerous research studies have been carried out on NNs. This is because they are powerful for performing complex tasks in a wide range of fields, such as system control,[3,4] communication,[5] and medical diagnosis.[6] We used a feedforward neural network (FFNN) based on a backpropagation (BP) learning algorithm with one hidden layer and 10 nodes.[14,15] When a sample $x_p = (x_{p1}, x_{p2}, ..., x_{pR})^T$ is input into the FFNN, it is distributed among the hidden layers, as shown by the structure in Fig. 2. The output o_{pj}^h of the jth node on the hth hidden layer with input net_{pj}^h can be calculated as

$$o_{pj}^h = f_j^h\left(net_{pj}^h\right) = \frac{1 - \exp\left(-2 \cdot net_{pj}^h\right)}{1 + \exp\left(-2 \cdot net_{pj}^h\right)},\tag{1}$$

in which $net_{pj}^h = \sum_{i=1}^{m} w_{j,i}^h x_{pi} + \theta_j^h$, with $w_{j,i}^h$ being the weight of the connection from the ith hidden layer to the jth hidden layer. θ_j^h and f_j^h denote the bias and transfer functions of the jth hidden node, respectively. Similarly, the output value of the kth output node can be calculated as
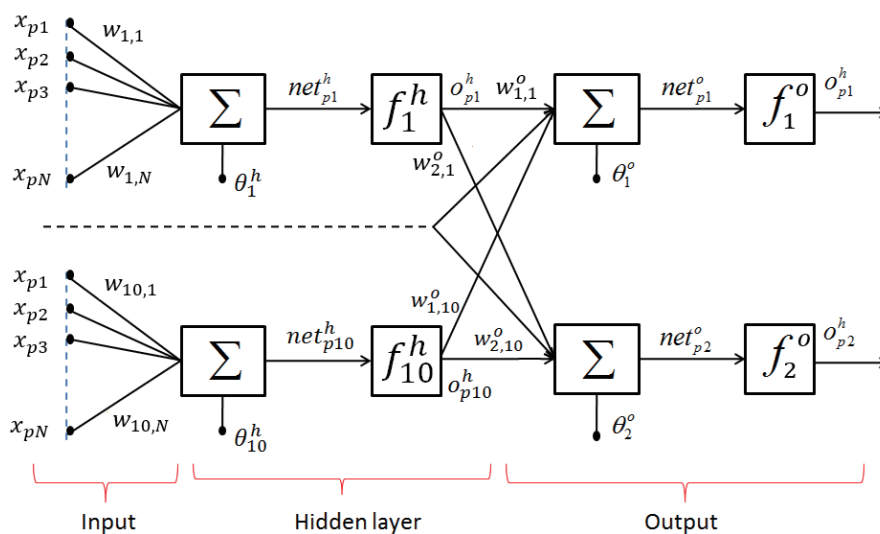

Fig. 2.   (Color online) Structure of whole NN.

$$o_{pk}^o = f_k^o \left(net_{pk}^o \right) = \frac{1}{1 + \exp\left(-net_{pk}^o\right)},$$ (2)

in which $net_{pk}^o = \sum_{i=1}^{m} w_{k,j}^o o_{pj}^h + \theta_k^o$, with $w_{k,j}^o$ being the weight of the connection from the jth hidden layer to the kth hidden layer. θ_k^o and f_k^o denote the bias and transfer functions of the kth output node, respectively. At the beginning, we assigned the initial weights and thresholds randomly, updating them at every iteration to minimize the difference or the mean square error E between the output and the target. The weights between the layers were updated in each iteration by the gradient-descent rule as follows:

$$w_{new} = w_{old} + \Delta w,$$ (3)

where $\Delta w_{k,j}^o = -\eta \dfrac{\partial E}{\partial w_{k,j}^o}$, $\Delta w_{k,i}^o = -\eta \dfrac{\partial E}{\partial w_{k,i}^o}$, and η is a step size in the range [0.01,1] in the formula for updating the weight from the jth layer to the kth layer and from the ith layer to the kth layer in the output node. We used 0.1 as the step size for 1000 iterations.

5. Performance Evaluation

One of the most popular methods of evaluating a model's prediction performance is cross-validation. There are two commonly used cross-validation methods, leave-one-out cross-validation (LOOCV) and k-fold cross-validation. We used k-fold cross-validation, more specifically, 10-fold cross-validation, because it has the advantage of using all samples in both training and validation. We partitioned all the samples randomly into 10 groups of the same size and used one group for testing and the others for training. We repeated the process until all the groups were tested, then all the results were averaged to a single estimation, which is called the true accuracy defined as

$$Acc = \frac{1}{K} \sum_{i=1}^{k} Acc_i.$$ (4)

Then, the accuracy factor, which represents the performance of the classifier, was estimated as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN},$$ (5)

in which *TP*, *TN*, *FP*, and *FN* denote true positive, true negative, false positive, and false negative, respectively.

6.   Results and Discussion

The NN-based classification system we built was trained using several different sets of features: only GLCM features, only GLRLM features, and both GLCM and GLRLM features. The classification results of training with GLCM features, GLRLM features, and both GLCM and GLRLM features obtained with the F-score feature selection method are shown in Figs. 3–5, respectively. There are a total of 52 GLCM features (Feature 1 to Feature 52) and a total of 44 GLRLM features (Feature 53 to Feature 96). We calculated and added features based on their F-score in descending order to train the network and compute the accuracy. Details of the feature extraction are given in Ref. 7. The F-score of the GLCM feature extraction method was as high as 0.225. When Feature 52 was added, the accuracy reached 80.75% (Fig. 3). For GLRLM, the highest F-score was 0.5 and the accuracy increased to 81.5% (Fig. 4) when Feature 96 was added, then decreased. By combining GLCM and GLRLM features, we can obtain
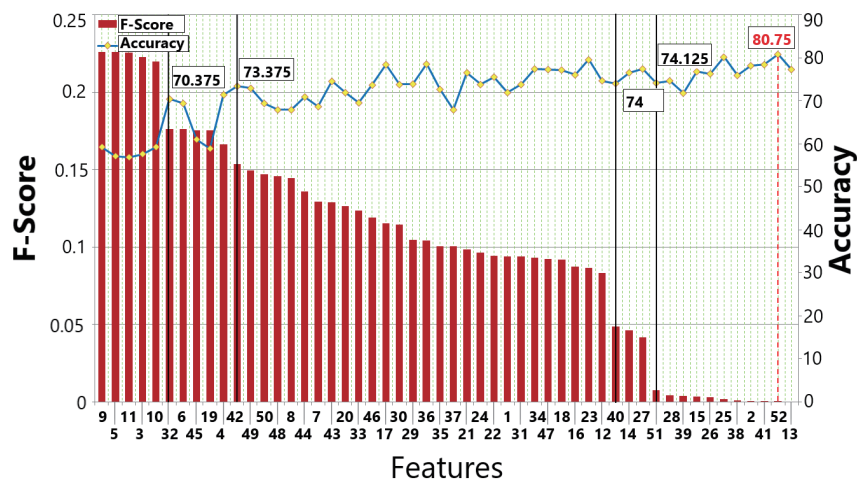

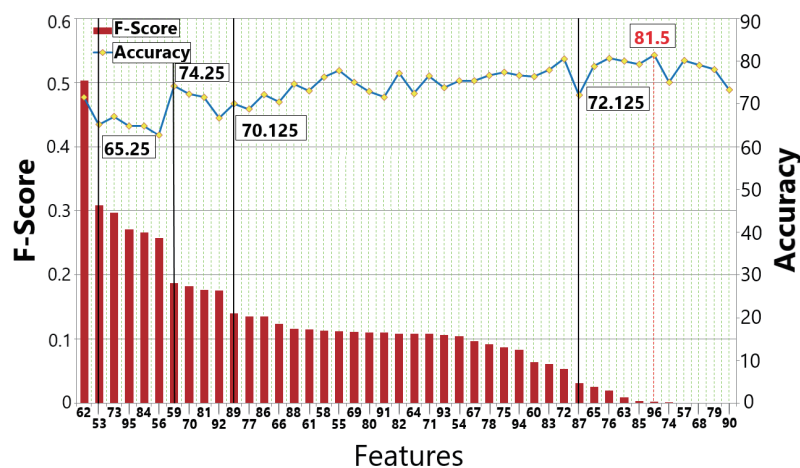Fig. 3.   (Color online) Results of NN-based classification using GLCM features.

Fig. 4.   (Color online) Results of NN-based classification using GLRLM features.
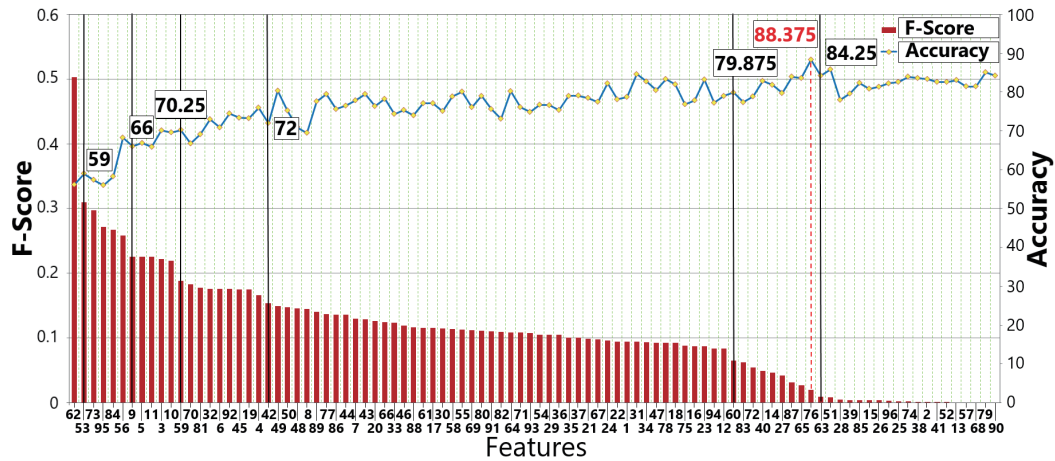
Fig. 5.   (Color online) Results of NN-based classification using GLCM and GLRLM features.

Table 3
Comparison of overall results.

	GLCM + NN	GLRLM + NN	All features + NN
No feature selection (%)	77.250	73.37	84.25
SFS (%)	72.250	63.00	71.25
SBS (%)	77.880	83.38	85.00
F-score (threshold) (%)	74.125	74.25	84.25
F-score (search all) (%)	80.750	81.51	88.375

an accuracy of up to 88.375% based on their F-scores up to Feature 76 (Fig. 5). Table 3 shows the classification results obtained using different feature selection methods. It can be seen that using feature selection models generally gives better results, except for SFS. The best result was obtained using the F-score search with the search-all method, which had an accuracy of 88.375%.

7.   Conclusion

Medical ultrasound is one of the diagnostic imaging techniques. It has several advantages, such as no radiation risk, low cost, easy operation, and non-invasiveness. On the other hand, a CAD system can help an inexperienced clinician in diagnostic evaluation. The novelty of this paper lies in introducing an NN-based classification system for ultrasound images with textural features to distinguish between HCC and liver abscess. We calculated GLCM and GLRLM feature matrices, and selected them by SFS, SBS, and F-score feature selection methods before using an NN to classify images. We verified its feasibility by employing an NN to classify HCC and liver abscess in this research. The proposed method can provide diagnostic help while distinguishing HCC from liver abscess with a high accuracy of up to 88.375%. A limitation of this study was the lack of a large amount of data for training and validation. As future research, an extended scheme for use with big data, such as that based on deep learning, can be considered.

Acknowledgments

References

1 S. C. Bushong: Diagnostic Ultrasound: Essentials of Medical Imaging Series (The McGraw Hill Company, USA, 1999).
2 M. Stocksley: Abdominal Ultrasound (Cambridge University Press, London, UK, 2001).
3 S. S.-D. Xu, H.-C. Huang, T.-C. Chiu, and S.-K. Lin: iRobotics **1** (2018) 28. http://140.120.90.160/iRobotics/page/pub/iRobotics_Vol_1_No_2_June_2018.pdf
4 S. S.-D. Xu, H.-C. Huan, T.-C. Chiu, and S.-K. Lin: Appl. Sci. **9** (2019) 1. https://doi.org/10.3390/app9051034
5 Y.-R. Chien, J.-W. Chen, and S. S.-D. Xu: IEEE Access **6** (2018) 21778. https://doi.org/10.1109/ACCESS.2018.2825239
6 C.-C. Lee, S.-T. Tsai, and C.-H. Yang: Sens. Mater. **30** (2018) 1859. https://doi.org/10.18494/SAM.2018.1899
7 S. S.-D. Xu, C.-C. Chang, C.-T. Su, and P. Q. Phu: Appl. Sci. **9** (2019) 1. https://doi.org/10.3390/app9020342
8 R. M. Haralick, K. Shanmugam, and I. H. Dinstein: IEEE Trans. Syst. Man Cybern. SMC-**3** (1973) 610. https://doi.org/10.1109/TSMC.1973.4309314
9 W. Gomez, W. C. A. Pereira, and A. F. C. Infantosi: IEEE Trans. Med. Imaging. **31** (2012) 1889. https://doi.org/10.1109/TMI.2012.2206398
10 X. Tang: IEEE Trans. Image Process. **7** (1998) 1602. https://doi.org/10.1109/83.725367
11 P. Pudil, J. Novovičová, and J. Kittler: Pattern Recognit. Lett. **15** (1994) 1119. https://doi.org/10.1016/0167-8655(94)90127-9
12 A. W. Whitney: IEEE Trans. Comput. **20** (1971) 1100. https://doi.org/10.1109/T-C.1971.223410
13 Y.-W. Chen and C.-J. Lin: Feature Extraction: Foundations and Applications, Guyon, I., Nikravesh, M., Gunn, S., Zadeh, L. A., Eds. (Springer, Berlin, 2006) Chap. 12.
14 M. Ridmiller: Comput. Standards Interfaces **16** (1994) 265. https://doi.org/10.1016/0920-5489(94)90017-5
15 R. D. Duda, P. E. Hart, and D. G. Stork: Pattern Classification (Wiley, New York, 2000) 2nd ed.