Towards an Explainable Machine Learning Framework for Sketched Diagram Recognition*

Amardeep Singh[1,*], Md Athar Imtiaz[2] and Rachel Blagojevic[3]

[1]*UCOL - Te Pūkenga, Palmerston North, New Zealand*
[2]*Massey University, Palmerston North, New Zealand*
[3]*Massey University, Palmerston North, New Zealand*

Abstract

In recent years, machine learning has made significant advancements in various fields, including image recognition. However, the complexity of these models often makes it difficult for users to understand the reasoning behind their predictions. This is especially true for sketch recognition, where the ability to understand and explain the model's decision-making process is crucial. To address this issue, our research focuses on developing an explainable machine learning framework for sketch recognition. The framework incorporates techniques such as feature visualization and feature attribution methods which provide insights into the model's decision-making process. The goal of this research is to not only improve the performance of sketch recognition models but also to increase their interpretability, making them more usable and trustworthy for users.

Keywords

Explainable AI, SHAP, Sketch recognition, Digital ink recognition, Diagram recognition

1. Introduction

The task of creating diagrams on a computer using a traditional mouse and keyboard can be a difficult task compared to the ease of drawing with a pen and paper. To bridge this gap, stylus-based devices are used to provide a similar user experience to paper-based sketching. Recognizing these sketches, or identifying elements in the drawing, can enhance the user experience by allowing for advanced functionalities such as automatic beautification, intelligent editing, and animation of the content. However, a challenge in the field of sketch recognition is maintaining high accuracy while still allowing for a free-sketch environment similar to traditional pen and paper. Even though recognition techniques have become more sophisticated, it is difficult to understand the inner workings of blackbox machine learning methods [1, 2, 3]. Without a deeper understanding, it is hard to make substantial improvements to the recognition algorithm's accuracy. In this research, we applied explainable AI techniques to assist in understanding how a machine learning based sketch recognition algorithm classifies instances. We believe this use of explainable AI (XAI) will lead to improved accuracy in future sketch recognition techniques. The main contributions of this work can be summarized as follow:

- **Inside the blackbox:** We are able to provide insights into the inner workings of a blackbox machine learning model for sketch recognition. By using techniques such as feature visualization and feature attribution methods like SHAP, we are able to provide a clear understanding of the model's decision-making process. This is important because it allows researchers to understand how the model is able to classify sketches and identify the important features that contribute to the predictions.
- **Methodology for understanding the blackbox:** We have outlined a methodology that can be used in future to understand blackbox sketch reocognisers. By incorporating interpretable models and feature attribution methods, we are able to provide a transparent and understandable explanation of the model's decision-making process. This is important because it allows users to trust the model's predictions and understand why they are being made.

This work is still in progress and the above contributions are important to the sketch community so that we can understand how and why blackbox algorithms work. By providing insights into the inner workings of the model, we are promoting trust in the algorithms and allowing researchers to improve the decisions being made by the recognisers. This is crucial for the further development and use of blackbox algorithms for sketch recognition.

Joint Proceedings of the ACM IUI Workshops 2023, March 2023, Sydney, Australia

[*] A. Singh, M. A. Imtiaz, and R. Blagojevic. 2023. Towards an Explainable Machine Learning Framework. In Joint Proceedings of the ACM IUI 2023 Workshops. Sydney, Australia, 7 pages

[*] Corresponding author.

a.singh@ucol.ac.nz (A. Singh); a.imtiaz@massey.ac.nz (M. A. Imtiaz); r.v.blagojevic@massey.ac.nz (R. Blagojevic)
0000-0003-1916-3347 (A. Singh)

The remainder of the paper is organised as follows. In section 2 we discuss related work. Section 3 describes our methodology. Section 4 presents the results of our experiments. Section 6 concludes the paper with directions for future work.

2. Related work

In recent years, there has been increasing interest in developing models that are not only accurate but also interpretable. The concept of model interpretability can be classified into two categories: global interpretability and local interpretability [4]. Global interpretability enables users to understand the overall structure of a model, while local interpretability focuses on the reasoning behind a model's decision for a specific input. Various techniques have been developed in Explainable Artificial Intelligence (XAI) to enhance model interpretability, including Attention Mechanisms [5], LIME (Local Interpretable Model-agnostic Explanations) [6], Saliency Maps [7], Counterfactual Analysis [8], Model Distillation [9], and SHapley Additive exPlanations (SHAP) [10] . SHAP has gained increasing attention as it provides both global and local interpretability by assigning an importance value to each feature in a prediction through the calculation of the average marginal contribution of the feature in all possible coalitions. It can measure feature importance for any model and handle interactions among features [10].

Previous research in XAI for blackbox machine learning-based sketch recognition algorithms has mainly focused on visualising Convolutional Neural Networks (CNN's), which is an image-based recognition approach. Peters et al. [11] produce videos using the dimensionality reduction method, UMAP, to visualise neuron activity in the training process. Mopuri et al. [12] are able to highlight discriminative regions of images classified by the CNN by examining the forward pass operation. Theodorus et al. [13] compare an interpretable model, BagNet, to blackbox CNN's. They use a score to rank the interpretability of a model based on heatmaps of discriminative regions of an image. Cai et al. [14] focus on end-user interaction with a sketch recognition system by providing two example-based explanations for predictions, normative and comparative. Normative explanations show examples from the target class (using the ground truth), while comparative explanations show examples of the closest predicted classes.

To our knowledge there have not been explorations into the use of XAI for other blackbox sketch recognition approaches, such as blackbox feature-based techniques e.g. using support vector machines [15] or ensembles [1]. While there are feature based approaches that are easier to interpret [16, 17], research directions have steered towards more sophisticated blackbox machine learning methods which produce higher accuracy rates [1]. However, although these blackbox algorithms might produce high recognition rates, interpreting the results and how the classifications are made s becoming far more difficult. The use of XAI techniques on such blackbox models can assist in interpreting results and therefore lead to the design of more successful sketch recognisers, as has been illustrated other areas of research such as healthcare [18].

3. Materials and methodology

This section presents the details of our proposed methodology.

3.1. Datasets

The chosen datasets are all full diagrams containing shapes and text together (as opposed to isolated shapes or text). They were chosen to represent a large variation of diagram domains, as we seek to investigate domain independent systems. They include examples of connected (e.g. directed graphs) and unconnected diagram domains (e.g. user interface). They also include variations in the placement of text, such as those with text inside shapes (e.g. organisation), outside shapes (e.g. Euler), or annotated connectors (e.g. process diagrams). Table 1 summarises datasets used in this research.

Table 1
Number of participants and strokes per dataset

Dataset	# Participants	# Text	# Shape	# Total
Training				
User interface [1]	20	4354	671	5025
Directed graph [1]	20	164	354	518
Organisation [1]	20	1098	607	1705
Verification				
ER [19]	33	2143	1050	3193
Process [19]	33	2674	1195	3869
Testing				
Mind-map [1]	20	1815	364	2179
To-do list [1]	20	1710	201	1911
UML class [1]	20	1481	383	1864
Euler [20]	9	60	60	120

3.2. Feature Library

For reliable and accurate recognition a set of quality features must be supplied to the algorithms. We employed Blagojevic et al's [21] digital ink feature library for our experiments. This library contains 114 features each measuring unique characteristics of each stroke such as curvature, density, direction, intersections, pressure, size, temporal and spatial context and time/speed.

3.3. Classification methods

We have used Extra-trees classifier, which is generally considered to be a black box technique because it uses an

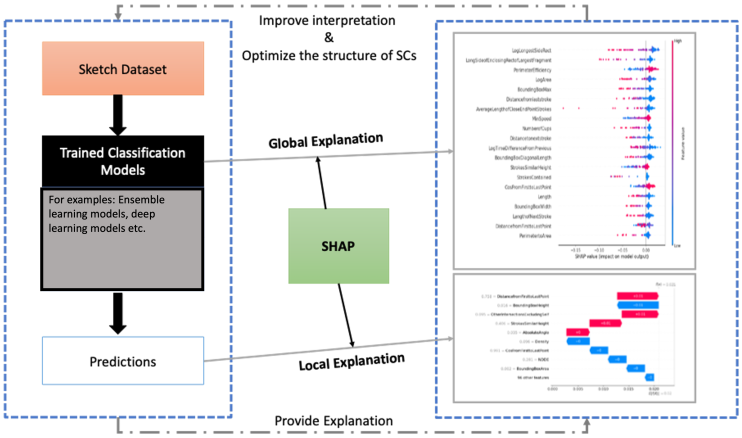
Improve interpretation
&
Optimize the structure of SCs
Sketch Dataset
Trained Classification Models
For examples: Ensemble learning models, deep learning models etc.
Predictions
Global Explanation
SHAP
Local Explanation
Provide Explanation

Figure 1: Overview of the structure of the proposed framework

ensemble of decision trees to make predictions. Decision trees are a type of supervised learning algorithm that is used to classify instances based on their features. It works by recursively partitioning the feature space into smaller regions, known as leaves, and making predictions based on the majority class within each leaf [22]. The Extra-trees classifier makes predictions based on a combination of features, and it may be difficult to determine which features are most important or how they are being weighted by the model [23].

3.4. Methodology

This section presents a framework for improving the interpretability of any sketch classification system (SCS). The framework is designed to enhance the transparency of SCS, which is crucial for human operators making decisions. The framework, as shown in Figure 1, comprises of two parts: the traditional structure of SCS on the left, and the interpretability-enhancing component on the right. The traditional structure includes the dataset, trained classification models, and predictions.

The focus of the interpretability framework is on providing local and global explanations, using the SHAP method, to improve experts' trust in the SCS. The main idea behind SHAP values is to calculate the contribution of each feature to the prediction of a specific sample. The SHAP value for a feature is defined as the average difference between the prediction of the model with that feature and the prediction of the model without that feature, for all possible coalitions of features. Mathematically, the SHAP value for a feature i for sample j denoted as $SHAP(i, j)$, is defined as:

$$SHAP(i,j) = \sum(S)\left[\frac{(|S|!)}{(|S| - |T|)!} \times |T|! \times \Big(f(S \cup \{i\}) - f(S)\Big)\right] \quad (1)$$

where, where $\sum(S)$ represents the summation over all possible sets of feature indices, S is a set of feature indices and T is a subset of S. $|S|$ represents the cardinality of set S, which is the number of elements in the set and $|S|!$ represents the factorial of $|S|$. $(|S| - |T|)!$ represents the factorial of $(|S| - |T|)$, which is the number of elements in the set S minus the number of elements in the subset T. $|T|!$ represents the factorial of $|T|$, which is the number of elements in the subset T. $f(S)$ represents the average prediction of the model when input features indexed by S are set to their baseline values. $f(SUi)$ represents the average prediction of the model when input features indexed by S are set to their baseline values and feature i is set to its actual value for sample j. $(f(SUi) - f(S))$ represents the difference between the average prediction of the model when input features indexed by S are set to their baseline values and feature i is set to its actual value for sample j, and the average prediction of the model when input features indexed by S are set to their baseline values.

In simple terms, equation 1 provides a local explanation i.e. provide an understanding of how each feature is impacting the prediction for a specific sample. Global explanations provide an understanding of the overall importance of each feature and how it impacts the predictions of the model across all samples. The global SHAP values are calculated for each feature and they represent the average change in the model output caused by setting feature i to its actual value, while holding all other features fixed at their baseline values across all samples. The equation for the global SHAP value of feature i is:

$$SHAP(i) = \sum(j)\Big[(f(SU\{i\}) - f(S))\Big] \quad (2)$$

In equation 2, j represents the index of the sample being evaluated, and S is a set of feature indices. The term $f(S)$ represents the average prediction of the model when input features indexed by S are set to their baseline values. On the other hand, $f(SUi)$ represents the average prediction of the model when input features indexed by S are set to their baseline values, and feature i is set to its actual value for the sample j.

4. Results and Discussion

This section describes the experimental setup, performance metrics used to evaluate the proposed approach and lastly, observed results are discussed in detail. This study was carried out using 2.3 GHz 8-core Intel i9 processor with 16 GB memory on Big Sur 11.4 operating system. The proposed approach is developed using Python programming language with several statistical and visualization packages such as Sckit-learn, Numpy, Pandas, Tensorflow, SHAP [24] and Matplotlib. In this work, we have used the Accuracy, Precision, Recall, F1-score for binary-class classification (text/shape).

Table 2
Classification results from extra-tree classifier

Dataset	Precison	Recall	F1-score	Accuracy
User Interface, Directed graph and Organisation diagrams dataset				**0.96**
Text	0.95	1.00	0.98	
Shape	1.00	0.83	0.91	
ER and Process diagrams dataset				**0.92**
Text	0.90	0.99	0.95	
Shape	0.98	0.77	0.87	
To-do list, Mind-Map and UML class diagrams dataset				**0.95**
Text	0.96	0.98	0.97	
Shape	0.89	0.78	0.83	
Euler diagrams dataset				**0.95**
Text	0.97	0.95	0.96	
Shape	0.94	0.96	0.95	

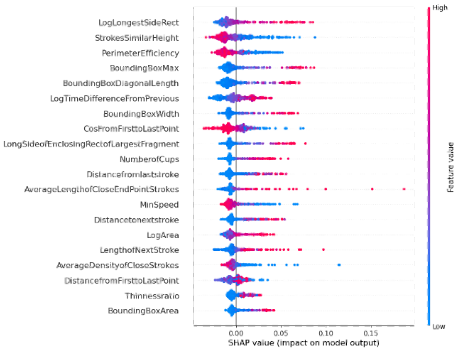

Figure 2: User interface, directed graph and organisation diagrams dataset

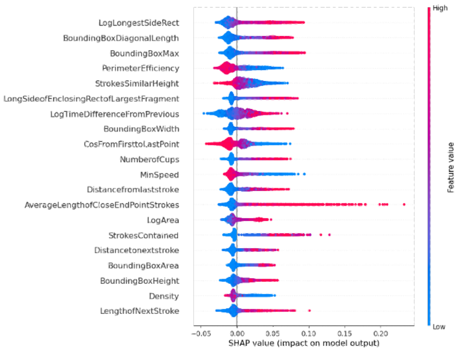

Figure 3: ER and Process diagrams dataset

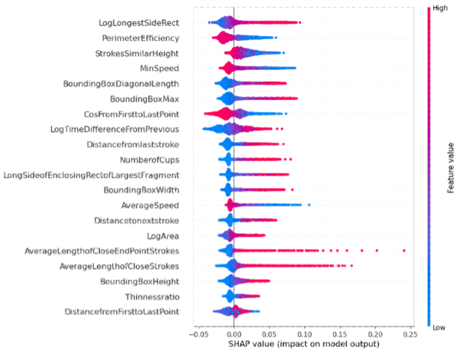

Figure 4: To-do list, Mind-map and UML class diagrams datasets

4.1. Discussion

We have made a number of different observations to understand the performance implications both during the training and testing phases. Table 2 presents the classification outcomes for various diagram datasets. The table shows the performance of the model on different datasets in terms of accuracy, precision, recall, and F1-score. From the table, it can be seen that the model performed well and was successful in identifying shape strokes and text strokes. Additionally, the table illustrates that text strokes had a higher recall rate compared to shape strokes, meaning that the model was able to correctly identify a higher percentage of text strokes than shape strokes out of all the text strokes that were present in the dataset. However, it is important to note that the accuracy, recall, precision, and F1-score can only provide an overall performance metric for the model and it does not explain the reasoning behind its decision-making. To understand the model's decision-making in more detail, the second part of the framework is used to provide global and local explanations of the model's predictions. Global explanations provide an understanding of the overall feature importance and how it impacts the predictions of the model across all samples. Local explanations provide an understanding of how each feature is impacting the pre-

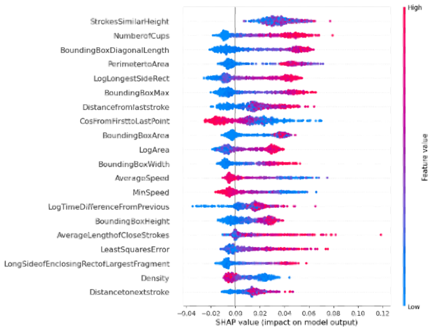

Figure 5: Mengli-Euler diagrams dataset

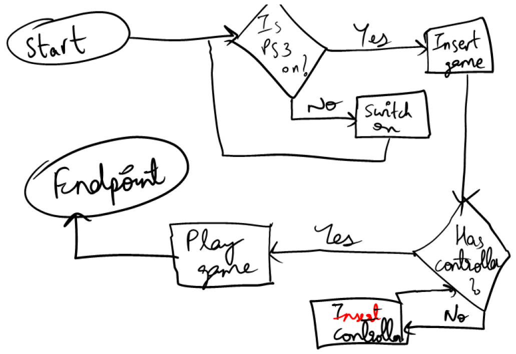

Figure 6: Misclassified text stroke

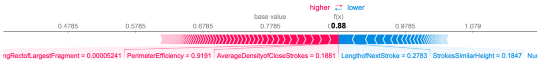

Figure 7: Local explanation visualisation behind wrong text sample

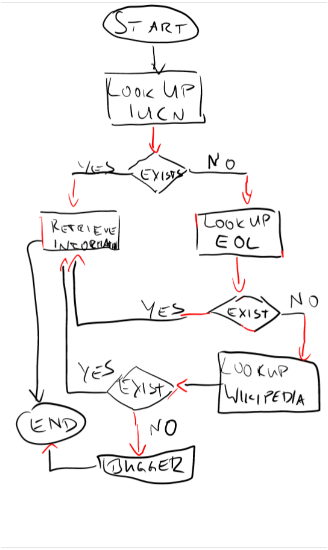

Figure 8: Misclassified shape strokes

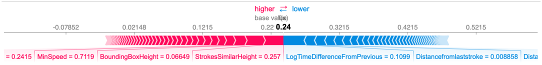

Figure 9: Local explanation visualisation behind wrong shape sample

diction for a specific sample. This can provide insights into which features are most important for the model's predictions and how the model is making its decisions. The beeswarm plots presented in Figure 2 through Figure 5 shows how each feature contributes to the overall output of a black-box model, providing a means to interpret the model's global explanations for each dataset. It is a combination of a scatter plot and a violin plot, where each dot represents a sample, and the y-axis represents the feature importance. To avoid overlapping, the dots in Figures 2 to 5 are horizontally jittered, and their colors represent the actual value of the feature for the corresponding sample; red dots indicate high feature values, while blue dots indicate low feature values. The violin plot in each figure shows the number of samples with similar feature values and can also identify outliers. These figures display the twenty most important features extracted from the extra tree classifier for the *shape* class in various datasets. Each point in the figures represents a Shapley value for a feature per sample, and the features are arranged in descending order of importance. For example, Figures 2 to 5 reveal that *LogLongestSideRect* is the top feature for the extra tree classifier, and the model will consider data points as shape if this feature has a larger value.

The local explanation is provided by Figure 7 and Figure 9 through a visual representation of the contribution of each feature to the model's predictions for individual samples. It also provides a reference point by showing the baseline which is the average prediction of the model when all features are set to their baseline values. It can be used to interpret the predictions of a black-box model and to identify any potential issues with the model's decision-making. The length of arrows tells about importance of feature in the prediction i.e. long arrows have a large effect on the prediction. These features are likely to be the most important for the model's decision. Features with short arrows have a small effect on the prediction. These features are likely to be less important for the model's decision. For example, Figure 6 shows text stroke classified as shape by the model. The Figure 7 shows an explanation behind wrong prediction. In this specific example, it can be seen that the *Average Density of Close Strokes* had a stronger influence on the model's decision to classify the stroke as a shape, while the *length of next stroke* had a stronger influence on the model's decision to classify it as text. In addition above two features are most important feature for model's decision for this particular instance. Similarly, Figure 8 show instances of arrows that were incorrectly classified as text, whereas Figure 9 providing an explanation for one arrow that was incorrectly classified as text.

5. Conclusion

This study aims to enhance the interpretability of sketch recognition models, as many machine learning models in this field do not provide any insight into the reasoning behind their decisions. Future work in this field could include the following. Firstly, XAI techniques could be applied to other types of models to better understand and interpret their predictions. In this work we focused on using XAI techniques to interpret and explain the predictions of a black-box ensemble learning model. However, there are many other types of models, such as deep neural networks and support vector machines that could also benefit from the use of XAI techniques. Secondly, using a combination of XAI techniques could be explored. Each XAI technique provides different types of explanations therefore combining them can further enhance our understanding of the model's decision-making by providing a more complete picture of the model's predictions. Lastly, building on Cai et als work [14], user feedback can be further incorporated to better understand how users interpret the explanations provided by the models.

References

[1] R. Blagojevic, B. Plimmer, J. Grundy, Y. Wang, Using data mining for digital ink recognition: Dividing text and shapes in sketched diagrams, Computers & Graphics 35 (2011) 976–991.

[2] D. Avola, M. Bernardi, L. Cinque, G. Foresti, C. Massaroni, Online separation of handwriting from freehand drawing using extreme learning machines, Multimedia Tools and Applications 79 (2020) 1–19. doi:10.1007/s11042-019-7196-1.

[3] M. Bresler, D. Prźša, V. Hlaváăź, Online recognition of sketched arrow-connected diagrams, Int. J. Doc. Anal. Recognit. 19 (2016) 253–267. URL: https://doi.org/10.1007/s10032-016-0269-z. doi:10.1007/s10032-016-0269-z.

[4] M. Du, N. Liu, X. Hu, Techniques for interpretable machine learning, 2019. arXiv:1808.00033.

[5] M. T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, Proceedings of the AAAI Conference on Artificial Intelligence 32 (2018). URL: https://ojs.aaai.org/index.php/AAAI/article/view/11491. doi:10.1609/aaai.v32i1.11491.

[6] M. T. Ribeiro, S. Singh, C. Guestrin, "why should i trust you?": Explaining the predictions of any classifier, 2016. URL: https://arxiv.org/abs/1602.04938. doi:10.48550/ARXIV.1602.04938.

[7] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, 2013. URL: https://arxiv.org/abs/1312.6034. doi:10.48550/ARXIV.1312.6034.

[8] S. Verma, V. Boonsanong, M. Hoang, K. E. Hines, J. P. Dickerson, C. Shah, Counterfactual explanations and algorithmic recourses for machine learning: A review, 2020. URL: https://arxiv.org/abs/2010.10596. doi:10.48550/ARXIV.2010.10596.

[9] Z. Che, S. Purushotham, R. Khemani, Y. Liu, Distilling knowledge from deep networks with applications to healthcare domain, 2015. URL: https://arxiv.org/abs/1512.03542. doi:10.48550/ARXIV.1512.03542.

[10] S. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems, 2017, pp. 4765–4774.

[11] M. Peters, L. Kempen, M. Nauta, C. Seifert, Visualising the training process of convolutional neural networks for non-experts., in: BNAIC/BENELEARN, 2019.

[12] K. R. Mopuri, U. Garg, R. Venkatesh Babu, Cnn fixations: An unraveling approach to visualize the discriminative image regions, IEEE Transactions on Image Processing 28 (2019) 2116–2125. doi:10.1109/TIP.2018.2881920.

[13] A. Theodorus, M. Nauta, C. Seifert, Evaluating CNN interpretability on sketch classification, in: W. Osten, D. P. Nikolaev (Eds.), Twelfth International Conference on Machine Vision (ICMV 2019), volume 11433, International Society for Optics and Photonics, SPIE, 2020, p. 114331Q. URL: https://doi.org/10.1117/12.2559536. doi:10.1117/12.2559536.

[14] C. J. Cai, J. Jongejan, J. Holbrook, The effects of example-based explanations in a machine learning interface, in: Proceedings of the 24th International Conference on Intelligent User Interfaces, IUI '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 258–262. URL: https://doi-org.ezproxy.massey.ac.nz/10.1145/3301275.3302289. doi:10.1145/3301275.3302289.

[15] M. Bresler, T. Van Phan, D. Prusa, M. Nakagawa, V. Hlavác, Recognition system for on-line sketched diagrams, in: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on, IEEE, 2014, pp. 563–568.

[16] D. Rubine, Specifying gestures by example (1991) 329–337. URL: http://doi.acm.org/10.1145/122718.122753. doi:10.1145/122718.122753.

[17] A. Bhat, T. Hammond, Using entropy to distinguish shape versus text in hand-drawn diagrams, in: IJCAI International Joint Conference on Artificial Intelligence, volume 9, 2009, pp. 1395–1400.

[18] G. Yang, Q. Ye, J. Xia, Unbox the black-box for the medical explainable ai via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond, Information Fusion 77 (2022) 29–52. URL: https://www.sciencedirect.com/science/article/pii/S1566253521001597. doi:https://doi.org/10.1016/j.inffus.2021.07.016.

[19] P. Schmieder, Comparing basic shape classifiers: A platform for evaluating sketch recognition algorithms, 2009.

[20] M. Wang, Exploring intuitive methods for creating euler and spider diagrams, 2011.

[21] R. Blagojevic, S. H.-H. Chang, B. Plimmer, The Power of Automatic Feature Selection: Rubine on Steroids, in: M. Alexa, E. Y.-L. Do (Eds.), Eurographics Workshop on Sketch-Based Interfaces and Modeling, The Eurographics Association, 2010. doi:10.2312/SBM/SBM10/079-086.

[22] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016. http://www.deeplearningbook.org.

[23] J. Younas, M. I. Malik, S. Ahmed, F. Shafait, P. Lukowicz, Sense the pen: Classification of online handwritten sequences (text, mathematical expression, plot/graph), Expert Systems with Applications 172 (2021) 114588. URL: https://www.sciencedirect.com/science/article/pii/S0957417421000294. doi:https://doi.org/10.1016/j.eswa.2021.114588.

[24] S. M. Lundberg, B. Nair, M. S. Vavilala, M. Horibe, M. J. Eisses, T. Adams, D. E. Liston, D. K.-W. Low, S.-F. Newman, J. Kim, et al., Explainable machine-learning predictions for the prevention of hypoxaemia during surgery, Nature Biomedical Engineering 2 (2018) 749.