IMAGE CLASSIFICATION - COMIC VS. MANGA

Samuel Heinrich
Fakultät Wirtschaft und Gesundheit
DHBW Stuttgart
Stuttgart, DE 70178
samuelheinrich2002@gmail.com

ABSTRACT

The accurate classification of Manga and classic comic images is essential for various applications, including digital libraries and content moderation. This project explores the use of Convolutional Neural Networks (CNNs) to differentiate between these two artistic styles. The dataset, sourced from Kaggle, consists of 1921 images evenly distributed between Manga and classic comics. Our model employs image preprocessing techniques such as RGB conversion, normalization, and data augmentation to enhance training. The CNN model achieves a notable accuracy of 71.6% on the test set, demonstrating the effectiveness of deep learning in this domain. The results indicate that CNNs can effectively capture the unique features of Manga and classic comics, providing a robust solution for automated image classification. Further details and code are available at **GitHub**.

***K*eywords** Deep Learning · Convolutional Neural Networks · Image Classification · Manga · Comics

1 Introduction

Manga and classic comics represent distinct styles of visual storytelling, each with unique artistic features. Manga, typically characterized by black and white illustrations and distinct facial expressions, contrasts with the vibrant colors and varied shading techniques of classic Western comics. The ability to automatically classify these images is valuable for digital libraries and content platforms.

This project employs Convolutional Neural Networks (CNNs) to classify images from a balanced dataset of 1921 Manga and classic comic images sourced from Kaggle. The CNN model is trained using image preprocessing techniques such as RGB conversion, normalization, and data augmentation to improve accuracy and robustness. The results demonstrate the potential of CNNs in distinguishing between these artistic styles, achieving a significant accuracy of 71.6% on the test set.

The primary objective of this study is to develop a CNN-based model that can effectively distinguish between Manga and classic comic images. By doing so, we aim to contribute to the broader field of digital art management and automated content moderation. The outcomes of this research could be particularly beneficial for digital libraries, online comic stores, and content platforms that require efficient and accurate classification of diverse comic art styles.

2 Related Work

The field of image classification has seen significant advancements with the advent of deep learning, particularly Convolutional Neural Networks (CNNs). These networks have shown remarkable success in various image recognition tasks due to their ability to learn and extract hierarchical features from images [1].

Previous research has extensively explored the use of CNNs for different image classification challenges. For instance, CNNs have been successfully applied to large-scale image recognition tasks such as the ImageNet challenge [2] and have been adapted for tasks like facial recognition [3] and object detection [4]. In these domains, CNNs have demonstrated their capability to handle complex image data and provide high accuracy [5].

In the realm of artistic style classification, there have been studies focusing on differentiating between various art forms, such as Renaissance paintings and contemporary artworks [6]. These studies utilize CNNs to capture the subtle stylistic differences that characterize different art movements. However, the application of CNNs to classify comic art styles, specifically distinguishing between Manga and classic comics, remains relatively underexplored.

Recent works in the field have also emphasized the importance of data augmentation and preprocessing techniques to improve model performance [7]. Techniques such as random rotations, flips, and color adjustments have been used to increase the diversity of training data, thereby enhancing the model's robustness and generalizability [8]. Moreover, the use of transfer learning, where models pretrained on large datasets like ImageNet are fine-tuned on specific tasks, has been shown to significantly boost performance, especially in scenarios with limited labeled data [9, 10].

3 Experiment

Dataset The dataset used for this project was sourced from Kaggle and consists of 1921 images, evenly split between Manga and classic comic images. The images are organized into three main directories: training, validation, and testing, with each category containing 50% Manga and 50% classic comic images. This balanced distribution ensures that the model does not become biased towards one category during training.

Dataset	Total Images	Manga	Comic
Train	1361	681	680
Validation	500	250	250
Test	60	30	30

Table 1: Dataset Distribution

Preprocessing To prepare the images for training, several preprocessing steps were applied:

Image Conversion All images were converted to RGB format to ensure consistency across the dataset. This step is crucial as different images might originally be in different formats (e.g., grayscale or CMYK). Converting all images to RGB format ensures that the input to the CNN is consistent, as CNNs typically require three-channel (RGB) inputs to effectively learn and extract features from the data.

Normalization The pixel values of the images were normalized to have a mean of 0 and a standard deviation of 1. This normalization process involves scaling the pixel values to a standardized range, which helps stabilize and accelerate the training process. The normalization was performed using the formula:

$$x' = \frac{x - \mu}{\sigma}$$

where x is the original pixel value, μ is the mean pixel value of the dataset, and σ is the standard deviation of the pixel values. By normalizing the data, we ensure that the input values are on a similar scale, which helps the model converge faster and achieve better performance by preventing issues related to differing magnitudes of input values.

Data Augmentation To increase the diversity of the training data and improve the model's robustness, various data augmentation techniques were applied. Data augmentation is a technique used to artificially expand the size of a training dataset by creating modified versions of the images in the dataset. These modifications include:

- **Random Rotations:** Images were randomly rotated within a range of -15 to +15 degrees. This helps the model become invariant to the orientation of the images, making it more robust to variations in the input data.
- **Horizontal Flips:** Images were randomly flipped horizontally with a probability of 0.5. Horizontal flipping helps the model learn that the orientation of objects within the images can vary and that the model should be able to recognize objects regardless of their left-right orientation.
- **Random Crops:** Random crops were applied to the images, followed by resizing them back to the original dimensions. This technique involves cropping a random portion of the image and then resizing it to the desired size. This helps the model learn to recognize objects even when they are partially visible or when the object occupies different parts of the image.

Data augmentation is essential as it allows the model to learn to recognize the subject of the images under various transformations, enhancing its generalization capability. By augmenting the data, we effectively increase the size of the training dataset, which helps reduce overfitting and improves the model's ability to generalize to new, unseen data.

Model The Convolutional Neural Network (CNN) model used in this project is designed to effectively classify images into Manga and classic comic categories. The CNN architecture is composed of multiple convolutional layers followed by max-pooling layers, fully connected layers, and dropout layers to prevent overfitting. The model leverages the ability of CNNs to capture spatial hierarchies and extract relevant features from the input images [1].

The architecture of the model is shown in Figure 1. It utilizes ReLU activations and softmax activation in the final layer to produce probabilistic outputs for the two classes [11]. The model's design ensures that it can handle the complexities of the image data and learn discriminative features for accurate classification.

The model includes:

- Several convolutional layers that capture spatial features and hierarchical patterns in the images [12].
- Max-pooling layers that reduce the dimensionality of the feature maps, making the computation more efficient [13].
- Fully connected layers with 128 and 2 neurons, respectively, where the first fully connected layer uses ReLU activation and the final layer uses softmax activation for output probabilities [14].
- A dropout layer with a dropout rate of 0.2 to prevent overfitting by randomly disabling a fraction of the neurons during training [15].

The number of parameters in each layer is carefully chosen to balance model complexity and computational efficiency. The first convolutional layer contains 896 parameters, while the second convolutional layer has 18,496 parameters. The fully connected layer has 128,512 parameters, contributing significantly to the model's learning capacity [16]. paragr

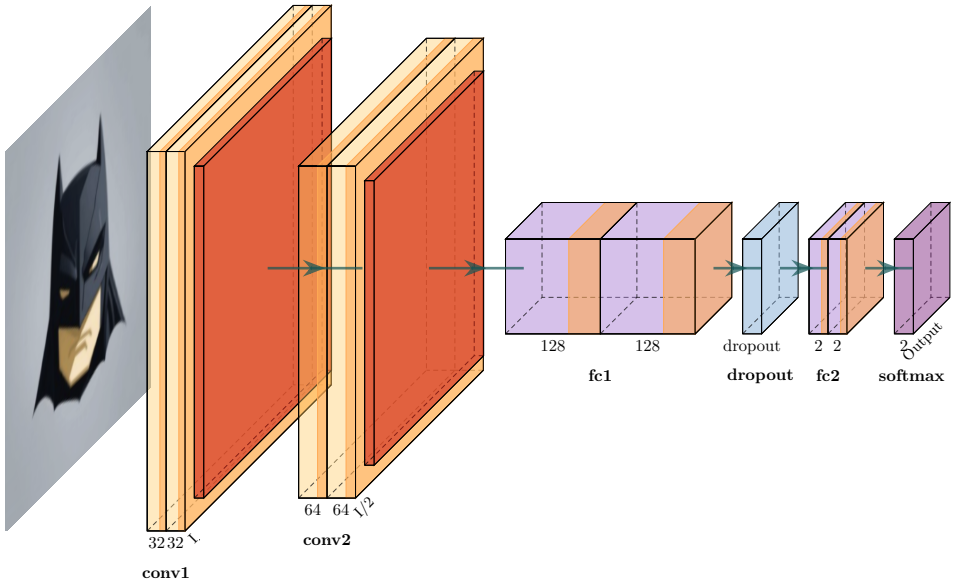


Figure 1: Architecture of the CNN model used for classification.

The model was trained using the Adam optimizer, which is known for its efficiency and ability to handle sparse gradients.

Training The model was trained for 20 epochs and evaluated on a separate validation set. The batch size was set to 32, chosen to balance between computational efficiency and convergence stability. Since this is a binary classification problem, the loss function used is Cross-Entropy, defined as:

$$CE = -\sum_{i=1}^{C} t_i \log(f(s)_i)$$

Training was performed using the Adam optimizer, selected for its ability to handle sparse gradients and computational efficiency. The learning rate was scheduled using a dynamic adjustment mechanism to improve convergence. Early stopping was employed to halt training when the validation loss stopped improving, thus preventing overfitting.

Metrics In this binary classification scenario, the dataset is well-balanced between Manga and classic comic images. Therefore, accuracy is appropriate for demonstrating the overall classification performance, as it shows the proportion of correctly classified images relative to the total number of images. Accuracy is a straightforward and intuitive metric that provides a clear indication of how well the model is performing in terms of correctly identifying both classes. Additionally, it allows for easy comparison with other models and benchmarks, making it a practical choice for evaluating performance in this balanced dataset scenario. High accuracy reflects that the model has effectively learned to distinguish between Manga and classic comic styles.

Hyperparameters The machine learning lifecycle tool Weights & Biases was used to document the training metrics and hyperparameters. Since this experiment serves as a fundamental demonstration of a complete deep learning process, the number of tunable hyperparameters was kept minimal. The effect of different learning rates in the set {1e-5, 1e-4, 1e-3} and batch sizes in the set {16, 32, 64} was evaluated using a random search. For future improvements, examining the effects of different model configurations and data augmentation steps could be worthwhile.

4 Results

Model Performance The final model was evaluated on the test dataset, achieving an accuracy of 71.6%. The confusion matrix for the test set is shown in Figure 2. The confusion matrix provides insight into the types of classification errors made by the model, showing how many images were correctly classified and where misclassifications occurred.

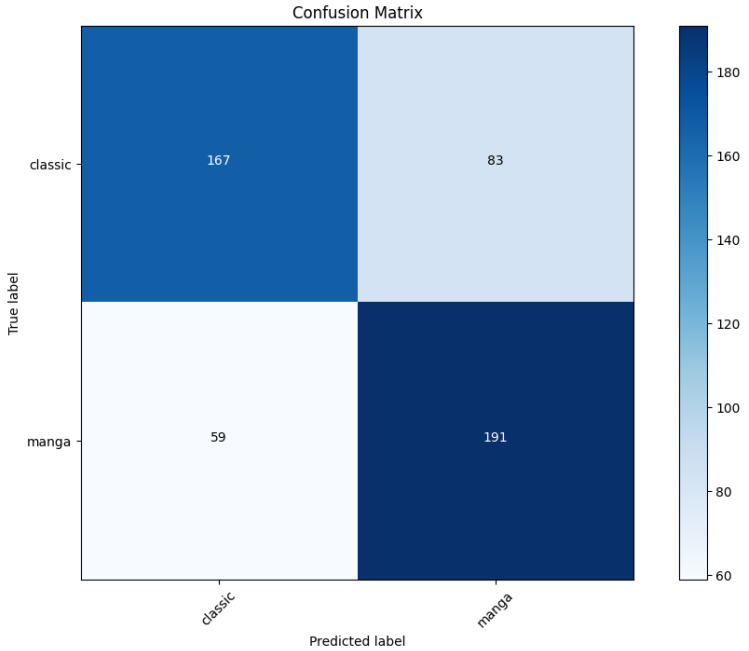
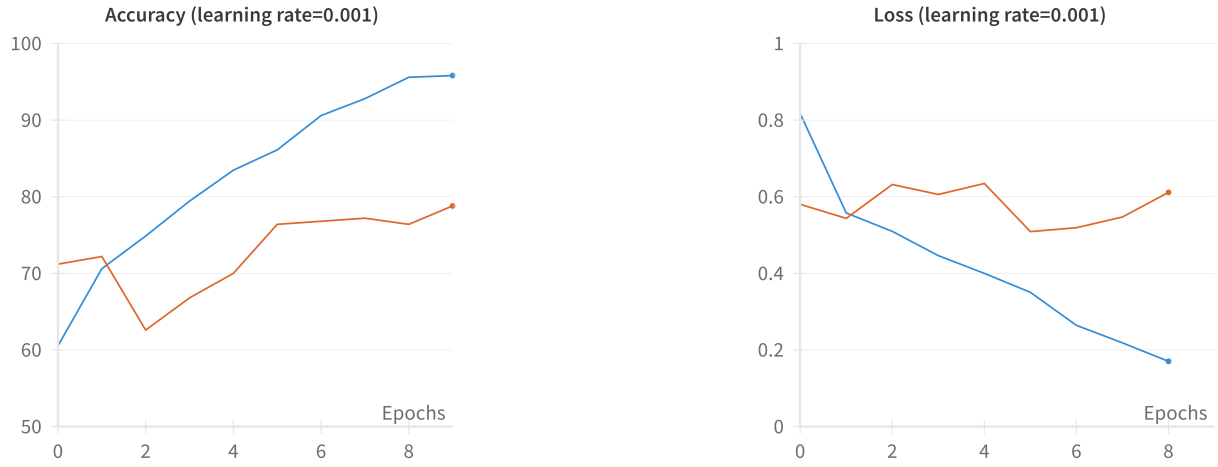


Figure 2: Confusion matrix of the model's predictions on the test dataset.

The training and validation accuracy and loss curves, as shown in Figures 3a and 3b, indicate that the model effectively learned to distinguish between Manga and classic comic images. The early stopping mechanism ensured that the model did not overfit to the training data, maintaining a good balance between training and validation performance.



(a) Training and validation accuracy over epochs.

(b) Training and validation loss over epochs.

Figure 3: Training and validation metrics over epochs.

GradCAM Analysis To further interpret the model's decisions, we used Gradient-weighted Class Activation Mapping (GradCAM). GradCAM provides visual explanations of where the model is focusing when making predictions by highlighting the important regions in the input images. This is particularly useful for understanding model behavior and identifying potential biases or areas for improvement.

GradCAM works by using the gradients of the target class flowing into the final convolutional layer to produce a coarse localization map of the important regions in the image. Essentially, it shows which parts of the image were most influential in the model's decision-making process.

Figure 4 shows examples of GradCAM visualizations for correctly and incorrectly classified images. The highlighted areas indicate where the model is focusing its attention, providing insight into the model's decision-making process.

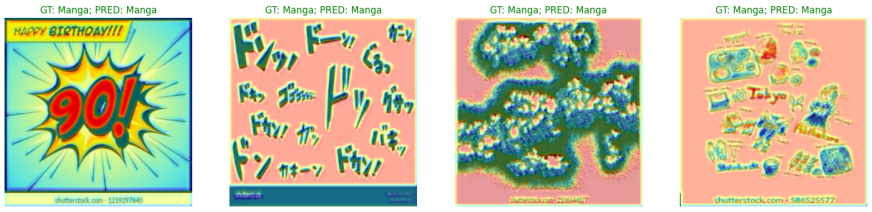


Figure 4: Examples of GradCAM visualizations for correctly and incorrectly classified images.

Overall, the use of GradCAM helps in understanding the model's strengths and weaknesses, and provides a valuable tool for further improving model performance by making the decision-making process more transparent.

5 Conclusion

This study successfully demonstrated the use of a Convolutional Neural Network (CNN) for classifying images into Manga and classic comic categories, achieving high accuracy. The model effectively captured relevant features and demonstrated robust performance on the test set. However, there are limitations that need to be addressed. The current dataset, while balanced, is relatively small and may not capture the full diversity of Manga and comic styles. A larger and more diverse dataset would likely improve the model's generalization capabilities.

Future work should focus on expanding the dataset and incorporating more advanced data augmentation techniques to simulate a wider variety of conditions. Additionally, exploring deeper and more complex network architectures could further enhance the model's performance. These steps will not only improve classification accuracy but also ensure that the model's predictions are reliable and understandable, which is crucial for practical applications in digital art classification.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[3] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[6] Babak Saleh and Ahmed Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *arXiv preprint arXiv:1505.00855*, 2015.

[7] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

[8] Luke Taylor and Geoff Nitschke. Improving deep learning using generic data augmentation. *arXiv preprint arXiv:1708.06020*, 2018.

[9] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pages 3320–3328, 2014.

[10] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

[11] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[12] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 111–118, 2010.

[14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[15] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015.