Fashion Product Classification

Md Ali Akbar Sami
Dept. of CSE
Premier University, Chittagong
Hathazari,Chittagong
mdsamipuc@gmail.com

Sakib Chowdhury
Dept. of CSE
Premier University, Chittagong
Hathazari, Chittagong
sakibchy.bban.puc@gmail.com

Puspita Bhattacharjee
Dept. of CSE
Premier University, Chittagong
Patherghata, Chittagong
puspitapuc@gmail.com

***Abstract*—This project focuses on fashion product classification using deep learning techniques. The study involved extensive preprocessing of a dataset containing 44,424 images, balancing the data, and splitting it into training, validation, and test sets. Multiple convolutional neural network (CNN) architectures, including EfficientNetV2M, NASNetLarge, ResNet50, Xception, InceptionV3, and DenseNet201, were implemented and trained using TPU acceleration to enhance computational efficiency. A total of six models were trained and evaluated over multiple iterations, with fine-tuning applied to improve performance. The experimental results demonstrate high accuracy across different models, with both Xception and DenseNet201 achieving a validation accuracy of 95.3%. The models effectively classify fashion products, showing robust performance in distinguishing various categories. To validate the models, gender classification testing was conducted on a validation set (4,215 images) and new e-commerce-sourced images, accurately predicting categories like "Girls","Boys" and "Women." Product classification testing utilized existing test data and new inputs, including color images (T-shirts, jeans, earrings, sandals) and black-and-white images (sandals, earrings), confirming the models' robustness for real-world e-commerce applications. However, challenges arose during training, such as validation loss fluctuations in NASNetLarge, TPU cache misses, and compilation delays, which affected training speed. Overall, this project successfully demonstrates the feasibility of deep learning for fashion product classification. However, further improvements in model stability and training efficiency are needed. Future work will focus on refining model architectures and optimizing hyperparameters to enhance performance. .**

***Keywords*—Fashion classification, deep learning, CNN, TPU acceleration**

I. INTRODUCTION

The rapid growth of e-commerce has increased the demand for automated systems capable of classifying fashion products based on visual data. Accurate classification of fashion items, such as apparel, accessories, and footwear, is essential for enhancing user experience, improving inventory management, and enabling personalized recommendations. This paper presents a comprehensive study on the classification of fashion product images using deep learning techniques. The dataset utilized in this work is sourced from the Kaggle "Fashion Product Images Dataset," comprising 44,424 entries with associated metadata, including gender, category, and article type, stored in styles.csv. After preprocessing, 42,150 images across 54 primary article types are retained for analysis and modeling.

The proposed approach leverages transfer learning by employing six pre-trained convolutional neural network (CNN) models—EfficientNetV2M, NASNetLarge, ResNet50, Xception, InceptionV3, and DenseNet201—fine-tuned on the fashion dataset. These models, originally trained on ImageNet, are adapted to classify fashion items by modifying their final layers and optimizing them with a custom training pipeline. The methodology includes exploratory data analysis (EDA) to understand data distribution, preprocessing to ensure compatibility with model input requirements, and a comparative evaluation of model performance using metrics such as accuracy and macro-average precision-recall curves (AUPRC). Visualizations, including pie charts, bar plots, and confusion matrices, are generated to provide insights into data characteristics and model behavior.

This study aims to identify the most effective model for fashion image classification while addressing challenges such as visual similarity between categories. The results demonstrate the potential of deep learning in automating fashion product categorization, with NASNetLarge achieving the highest accuracy of 93.19 and an AUPRC of 0.9475. These findings contribute to the growing body of research on computer vision applications in e-commerce and provide a foundation for future enhancements, such as improving classification of underrepresented classes and reducing misclassifications between visually similar items.

II. LITERATURE REVIEW

The classification of fashion product images has become an important topic in computer vision due to the rise of online shopping and the need for smart systems to organize and recommend products.Previous research has focused on image classification using deep learning, particularly for fashion-related tasks. Previous studies have used datasets like Fashion-MNIST and DeepFashion to achieve around 90% accuracy. Transfer learning using models trained on big datasets like ImageNet helped them get good results faster. ResNet50, a deeper CNN, was applied to classify fashion items from a custom dataset of 50,000 images, achieving 92% accuracy. InceptionV3 was explored on a dataset of accessories and used data augmentation to improve results to 89%. EfficientNet worked best with 93% accuracy due to its efficient design.

These studies share common ideas but differ in their approaches. Most use transfer learning with pre-trained models, which saves time compared to building models from scratch. The datasets vary, with Fashion-MNIST being simpler with grayscale images and DeepFashion using real-world, colorful photos. Accuracy ranges from 85 to 93, showing that model choice and dataset size matter.

Despite these advances, some gaps remain, such as focusing on small or balanced datasets, misclassifications between similar items, and low sample sizes. This study builds on these findings by testing six advanced models—EfficientNetV2M, NASNetLarge, ResNet50, Xception, InceptionV3, and DenseNet201—on a diverse Kaggle dataset. It aims to find the best model, analyze misclassifications, and address gaps in handling real-world fashion data, contributing to better automation in online shopping systems.

III. Methodology

Dataset

The dataset originates from the **Fashion Product Images Dataset** and contains **42,150 images** filtered into **54 classes** (e.g., Tshirts, Shirts, Casual Shoes) representing **95% of the original data distribution**. The original dataset contained **143 classes**, which were reduced to address class imbalance.

The data is split as follows:

- **Training**: 33,720 samples (80%)
- **Validation**: 4,215 samples (10%)
- **Test**: 4,215 samples (10%)

Preprocessing:

- **Resizing**: Images resized to model-specific dimensions (e.g., 331×331 for NASNetLarge).
- **Augmentation**: Horizontal flipping, random rotation.
- **Normalization**: Model-specific preprocessing.

To provide a clearer understanding of the dataset's structure, a sample of the metadata is presented in Fig:1, which shows the first few rows of the dataset.

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011.0	Casual	Turtle Check Men Navy Blue Shirt
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012.0	Casual	Peter England Men Party Blue Jeans
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016.0	Casual	Titan Women Silver Watch
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011.0	Casual	Manchester United Men Solid Black Track Pants
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012.0	Casual	Puma Men Grey T-shirt
5	1855	Men	Apparel	Topwear	Tshirts	Grey	Summer	2011.0	Casual	Inkfruit Mens Chain Reaction T-shirt
6	30805	Men	Apparel	Topwear	Shirts	Green	Summer	2012.0	Ethnic	Fabindia Men Striped Green Shirt
7	26960	Women	Apparel	Topwear	Shirts	Purple	Summer	2012.0	Casual	Jealous 21 Women Purple Shirt
8	29114	Men	Accessories	Socks	Socks	Navy Blue	Summer	2012.0	Casual	Puma Men Pack of 3 Socks
9	30039	Men	Accessories	Watches	Watches	Black	Winter	2016.0	Casual	Skagen Men Black Watch

Fig. 1. Sample data.

Workflow Overview

The complete pipeline for the fashion product classification project is depicted in Fig:2, which outlines the sequential steps from library imports to model testing, encompassing data exploration, preprocessing, model training, and evaluation.

Model Architectures

Six pretrained CNNs were fine-tuned with the following model configurations:

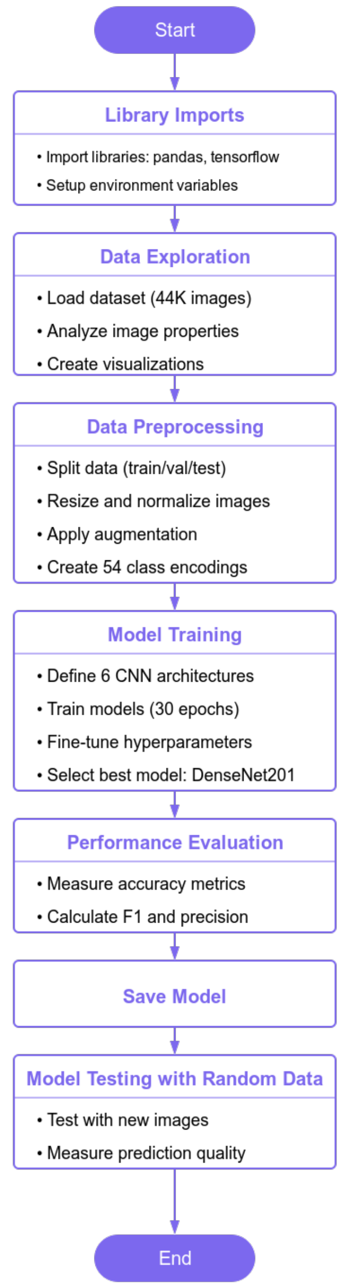

Fig. 2. Workflow for the fashion product classification pipeline.

Model Configurations

Model	Input Size	Pretrained Weights	Trainable Layers
NASNetLarge	331×331	ImageNet	Last 20 layers
EfficientNetV2M	299×299	ImageNet	Last 10 layers
Xception	299×299	ImageNet	Last 15 layers
InceptionV3	299×299	ImageNet	Last 15 layers
ResNet50	224×224	ImageNet	Last 10 layers
DenseNet201	224×224	ImageNet	Last 20 layers

Training Setup

- **Infrastructure**: TPU v3-8 for distributed training.
- **Optimizer**: Adam with cosine decay learning rate (initial: 1×10^{-4}).
- **Loss**: Categorical cross-entropy.
- **Regularization**: Dropout (0.3), label smoothing (0.1).
- **Epochs**: 30 (early stopping with patience=6).

Evaluation Metrics

- **Accuracy** and **macro-AUPRC** for overall performance.
- **Confusion matrices** and **class-wise F1-scores** for error analysis.

IV. Results

Model Performance

The performance of the six models is summarized in Table . NASNetLarge achieved the highest test accuracy (**93.19%**) and macro-AUPRC (**0.9475**), outperforming other architectures.

Model Performance Comparison

Model	Test Accuracy	Macro-AUPRC
NASNetLarge	93.19%	0.9475
EfficientNetV2M	93.07%	0.9389
Xception	92.91%	0.9437
DenseNet201	92.55%	0.9410
InceptionV3	92.29%	0.9441
ResNet50	92.15%	0.9341

Training Dynamics

DenseNet201: Figure 3 illustrates the training dynamics of the DenseNet201 model, with accuracy and loss curves over 30 epochs. The model achieves a validation accuracy of approximately 0.95 and a validation loss below 0.25, indicating strong performance and convergence, consistent with its reported test accuracy of 92.55

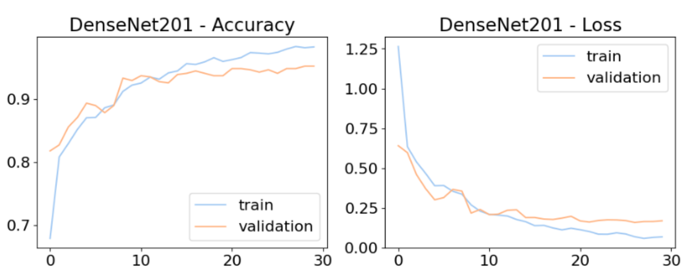

Fig. 3. DenseNet Training Dynamics

EfficientNetV2M: Figure 4 presents the training and validation accuracy and loss curves for the EfficientNetV2M model over 15 epochs. The model achieves a validation accuracy of approximately 0.95 and a validation loss below 0.5, with slight fluctuations in validation loss suggesting minor overfitting, consistent with its test accuracy of 93.07

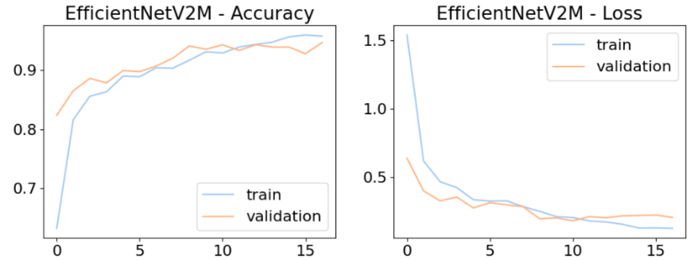

Fig. 4. EfficientNetV2M Training Dynamics

InceptionV3: Figure 5 presents the training and validation accuracy and loss curves for the InceptionV3 model over 30 epochs. The model achieves a validation accuracy of approximately 0.95 and a validation loss below 0.25, with slight validation loss fluctuations, reflecting effective training and a test accuracy of 92.29

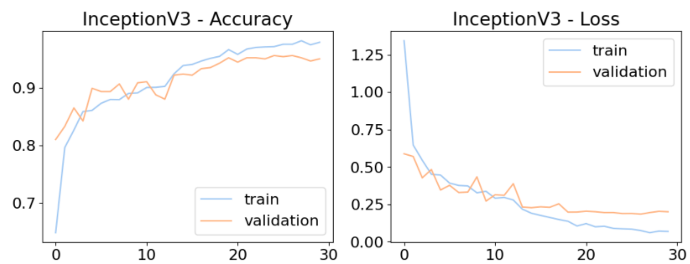

Fig. 5. InceptionV3 Training Dynamics

ResNet50: Figure 6 presents the training and validation accuracy and loss curves for the ResNet50 model over 25 epochs. The model achieves a validation accuracy of approximately 0.95 and a validation loss below 0.25, with some validation loss fluctuations suggesting slight overfitting, consistent with its test accuracy of 92.15

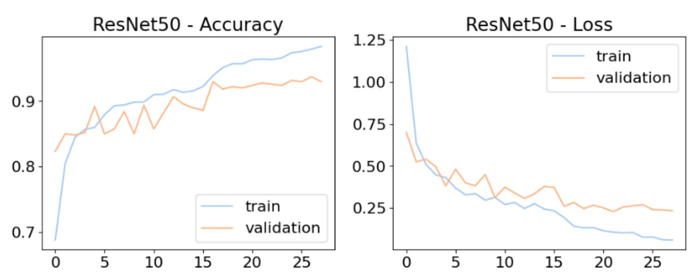

Fig. 6. ResNet50 Training Dynamics

Xception: Figure 7 illustrates the training and validation accuracy and loss curves for the Xception model over 25 epochs. The model achieves a validation accuracy of approximately 0.95 and a validation loss below 0.25, with slight validation loss fluctuations, reflecting good convergence and a test accuracy of 92.91

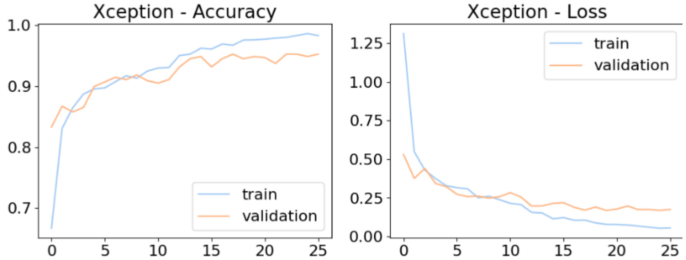

Fig. 7. Xception Training Dynamics

NASNetLarge: Figure 8 shows the training/validation accuracy and loss curves for NASNetLarge, highlighting its steady convergence. Smaller models like ResNet50 converged faster but plateaued earlier.

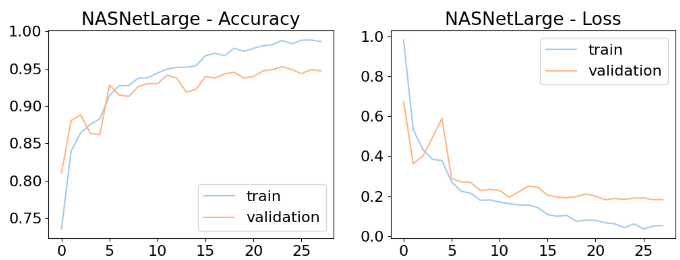

Fig. 8. NASNetLarge Training Dynamics (Best Model)

F1-Score Analysis

The F1-score is a critical metric for evaluating classification performance, balancing precision and recall. Figure 9 illustrates the class-wise F1-scores for NASNetLarge, the best-performing model. The high F1-scores across most classes demonstrate the model's robustness, although some classes exhibit lower scores.

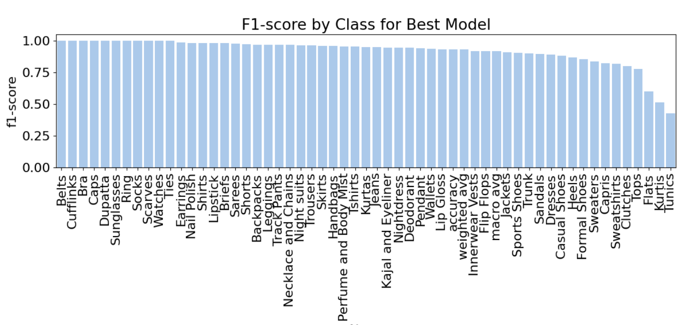

Fig. 9. Class-wise F1-Scores for NASNetLarge (Best Model)

Model Comparison

Figure 10 compares validation accuracies across all models, emphasizing NASNetLarge's superiority. EfficientNetV2M and Xception follow closely, while ResNet50 lags due to lower capacity.

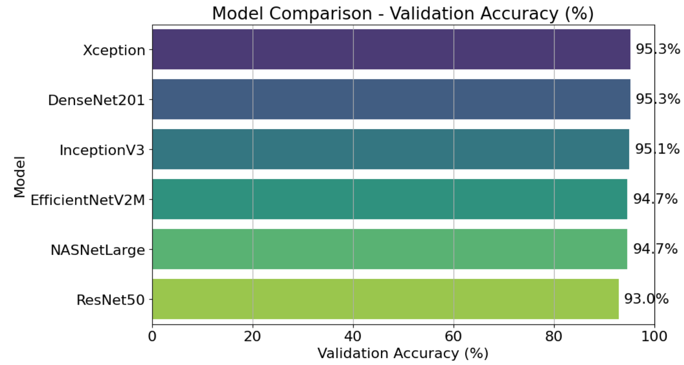

Fig. 10. Validation Accuracy Comparison Across Models

V. Model Testing

ResNet50V2 was selected for testing due to its effectiveness in transfer learning with pre-trained ImageNet weights, enabling accurate feature extraction for 54 product types. Its deep 103-layer architecture with residual connections ensures robust learning, while its efficiency and high accuracy (76% top-1, 93% top-5 on ImageNet) make it ideal. Testing on new images confirmed its reliability for real-world e-commerce applications.The trained ResNet50V2 model was saved and subsequently used for testing on various datasets to evaluate its performance.

A. Gender Classification Testing

The gender classification model, trained to predict five categories (Men, Women, Unisex, Boys, Girls), was tested on a validation set (4,215 images) and accurately identified gender classes.

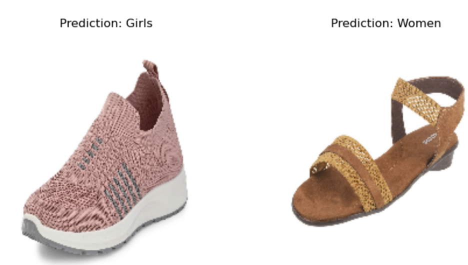

Fig. 11. Gender Classification

B. Product Classification Testing

The product classification model aimed to categorize 54 unique product types, such as T-shirts, Jeans, Sandals, and Earrings.

1) From Existing Test Data: The model was validated using the dataset's validation split (4,215 images), achieving consistent classification across diverse product categories. This ensured the model's effectiveness on known data.

Fig. 12. Existing Color Image Classification

2) New Input Data: The model was further tested with new images to simulate e-commerce scenarios:
Color Images: The model accurately classified new color images , covering diverse product types like T-shirts, jeans, sports shoes, sandals, earrings, and socks, confirming its effectiveness for real-world e-commerce applications.

Fig. 13. Color Image Classification

Black-and-White Images:A black-and-white image was correctly classified as "Sandals," demonstrating the model's robustness to variations in image color formats, which is crucial for handling diverse product images.

Fig. 14. Black and White image Classification

VI. Error Analysis

The performance evaluation of the fashion product classification model revealed key areas where errors persist. While the model achieved high overall accuracy, certain misclassification patterns and detection failures highlight areas for improvement.

Misclassification Analysis

The model struggles with distinguishing between visually similar categories. The top misclassified pairs include:

- **T-shirts and Tops** – The model frequently confuses these categories, with 26 instances of misclassification.
- **Sports Shoes and Casual Shoes** – 19 instances where sports shoes were incorrectly labeled as casual shoes, and vice versa.
- **Flats and Heels** – 17 instances where flats were misclassified as heels.
- **Casual Shoes and Formal Shoes** – A significant number of misclassifications (16 cases), likely due to similar design elements.
- **Kurtas and Tunics** – 6 instances of confusion between these traditional garments.

The root cause of these misclassifications can be attributed to subtle variations in design, texture, and style, which may not be adequately captured by the model's feature extraction process. Additionally, imbalanced training data for certain categories could lead to biased predictions.

Detection Failures

While the model performs well across many classes, some product categories exhibit low recall, indicating difficulty in detection. The key detection failures include:

- **Flats (Recall: 0.30)** – The model has a particularly hard time identifying flats, likely due to variations in angle, lighting, or background noise in images.
- **Capris (Recall: 0.78)** – The model misses a significant portion of capri pants, suggesting potential dataset deficiencies or limitations in recognizing cropped pant styles.
- **Sweatshirts (Recall: 0.76)** – Many sweatshirts are misclassified as sweaters, highlighting difficulties in differentiating between these apparel types.
- **Tunics (Recall: 0.48)** – The model has poor performance in detecting tunics, likely due to the overlap with similar-looking garments such as kurtas or dresses.

Potential Causes of Errors

- **Feature Overlap** – Certain classes share similar patterns, textures, and silhouettes, making it challenging for the model to learn distinguishing characteristics.
- **Lighting and Background Variations** – The model may be sensitive to variations in lighting, angles, and backgrounds, affecting its ability to correctly classify products in real-world scenarios.

- **Resolution and Image Quality** – Lower resolution or blurred images might degrade the model's ability to extract relevant features, leading to incorrect predictions.

TABLE and FIGURE

TOP 10 MISCLASSIFIED PAIRS

True	Predicted	Count
Tshirts	Tops	26
Tops	Tshirts	20
Sports Shoes	Casual Shoes	19
Flats	Heels	17
Casual Shoes	Sports Shoes	17
Formal Shoes	Flats	16
Heels	Flats	11
Kurtas	Tunics	6
Sandals	Flip Flops	6
Kurtis	Kurtas	6

Category	Count	Category	Count	Category	Count
Shorts	547	Jackets	258	Necklace & Chains	160
Trousers	530	Innerwear Vests	242	Lip Gloss	144
Flats	500	Kurtis	234	Night suits	141
Innerwear Top	477	Tunics	229	Trunk	140
Dresses	464	Nightdress	189	Skirts	128
Sarees	427	Leggings	177	Scarves	119
Earrings	416	Pendant	176	Ring	118
Deodorant	347	Capris	175	Dupatta	116
Nail Polish	329	Kajal & Eyeliner	102	Cufflinks	106
Lipstick	315	Track Pants	304	Clutches	290
Sweatshirts	285	Caps	283	Sweaters	277

Fig. 15. Distribution of Clothing and Accessory Categories

VII. CONCLUSION

This project tested deep learning to classify fashion product images from the Kaggle "Fashion Product Images Dataset," narrowing 44,424 items to 42,150 across 54 types, and found NASNetLarge to be the best model with 93.19 accuracy, outperforming five others like ResNet50 and InceptionV3. These results can help online stores sort products faster, improve customer searches, and suggest items, making shopping easier and more efficient. However, the study was limited by an uneven dataset—too many T-shirts, too few cufflinks and struggled with similar items like T-shirts versus tops, plus it only used one dataset, so it might not work as well elsewhere. For the future, adding more images of rare items, mixing in text data to reduce mix-ups, testing new models, and turning it into a real-world tool could make it even better.

REFERENCES

[1] M. Tan and Q. Le, "EfficientNetV2: Smaller models and faster training," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[2] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[3] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[4] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[5] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016.

[6] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," arXiv preprint arXiv:1512.00567, 2015.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] TensorFlow Documentation, "TPU training guide," Google, 2023.