Fashion-MNIST Image Classification using CNN and Grad-CAM

Yoojun Kim
Construction Science
yoojun@tamu.edu

Boyu Li
Electrical & Computer Engineering
liboyu1999@tamu.edu

Changju Bak
Electrical & Computer Engineering
cjbark1225@tamu.edu

***Abstract*—This project presents the development and evaluation of a Convolutional Neural Network (CNN) model for image classification, utilizing the Fashion-MNIST dataset. An in-depth Exploratory Data Analysis (EDA) of this dataset is first conducted, highlighting significant variations in pixel sizes across different classes and the associated challenges in classification. The CNN model, enhanced with Gradient-weighted Class Activation Mapping (Grad-CAM), is designed, and hosted on Google Colab to ensure robustness and reproducibility. Notably, our model achieves a 91.2% accuracy rate on the Fashion-MNIST test dataset, signifying relatively its competitive performance compared to state-of-the-art models. Furthermore, the incorporation of Grad-CAM enhances our model's capabilities in eXplainable Artificial Intelligence (XAI), offering insightful visual explanations for the classification decisions, especially in differentiating similar categories such as 'shirt' and 'coat'. The study also delves into the practical applications of this model in fashion design and marketing, discussing its potential benefits and limitations in these domains.**

***Keywords*—Classification, CNN, Grad-CAM, Fashion-MNIST**

I. Introduction

This project aims to design and evaluate a Convolutional Neural Network (CNN) as an image classifier, utilizing the Fashion-MNIST dataset. CNNs, exemplified by models like AlexNet [1], Faster R-CNN [2], and ResNet-50 [3], were extensively employed for their proven efficacy in similar applications [4], [5]. However, the inherent complexity of CNNs often results in a 'black box' model, where the reasoning behind decisions remains obscure. To enhance transparency, this project incorporates Gradient-weighted Class Activation Mapping (Grad-CAM) [6], a technique that visually demystifies the CNN's decision-making by highlighting key regions in images that influence predictions. This approach not only offers a deeper understanding of the model's predictive mechanics but also bridges the gap between high performance and interpretability in deep learning. Following the project instruction, the project is methodically structured, encompassing data preparation, Exploratory Data Analysis (EDA), model selection and training/evaluation, culminating in a discussion on the interpretability and practical applications of the model.

II. Method

A. Exploratory Data Analysis (EDA)

Figure 1 shows the dataset consists of 10 distinct classes, with each class containing 6000 samples for training set and 1000 sample for testing set. This indicates a balanced

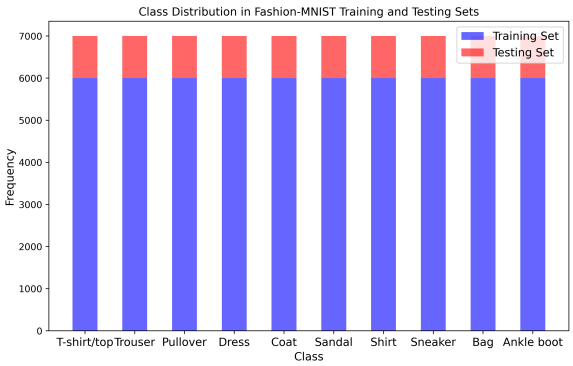


Fig. 1. Class frequency in Fashion-MNIST training and testing sets

distribution in terms of image sample frequency across all classes. Prior to in-depth Exploratory Data Analysis (EDA), we normalized the images by dividing them by 255, scaling the values to a range of 0 to 1 of both data sets. Our in-depth EDA comprises two key elements: a statistical analysis of instance sizes across classes and a visualization technique employing dimensionality reduction to highlight intra-class feature similarities.

In our study, we analyze the distribution of pixel ratios for each image per class in both training and testing sets, recognizing that the variance and size of instance pixels are key indicators of a dataset's complexity and challenges. We calculate the ratio of non-background pixels (instance pixels) for each image across both datasets. To ascertain statistical differences per class between the datasets, we conduct a one-way ANOVA test. The outcomes of these tests are visually represented in Figure 2, showcasing the ratio distributions for each dataset. Notably, classes like 'sneaker' exhibit smaller mean and variance in pixel ratios, whereas 'sandal' and 'bag' display larger variances. A significant observation, as presented in Table 1, is the statistical discrepancy in the instance size distribution of the 'coat' class between the two datasets (P-value=0.006). This finding suggests potential imbalances in instance size distributions, highlighting an area that may require further investigation to understand dataset representativeness and model training effectiveness.

Next, we employ the Uniform Manifold Approximation and

TABLE I
DESCRIPTIVE ANALYSIS AND ONE-WAY ANOVA BASED ON INSTANCE PIXEL RATIO ON THE FASHION-MNIST TRAINING AND TESTING SETS

Class	Training Sets			Testing Sets			One-way ANOVA	
	Median	Mean	Variance	Median	Mean	Variance	F-Value	P-Value
T-shirt/top	0.599489796	0.59412415	0.005745762	0.599489796	0.594985969	0.005972149	0.110148015	0.739985625
Trouser	0.332908163	0.348676871	0.004796205	0.334183673	0.352626276	0.005128775	2.759394417	0.096729625
Pullover	0.646045918	0.64871875	0.004984733	0.645408163	0.652030612	0.005060764	1.881423086	0.170216467
Dress	0.427295918	0.427908163	0.008696892	0.428571429	0.433053571	0.009001392	2.595600516	0.107205914
Coat	0.607142857	0.60173108	0.005623767	0.614795918	0.608727041	0.005420182	7.496323637	0.006198139
Sandal	0.31505102	0.321141369	0.01090559	0.316964286	0.322024235	0.010522896	0.061553365	0.804064697
Shirt	0.635204082	0.629123299	0.006231399	0.635204082	0.627424745	0.006966227	0.390164005	0.53223357
Sneaker	0.329081633	0.337927509	0.002920835	0.327806122	0.33783801	0.002826597	0.002360794	0.96124902
Bag	0.586734694	0.585614796	0.012982208	0.589285714	0.588728316	0.013573365	0.635723085	0.425290838
Ankle boot	0.477040816	0.484519983	0.005694722	0.474489796	0.483603316	0.005679408	0.126487138	0.722113186

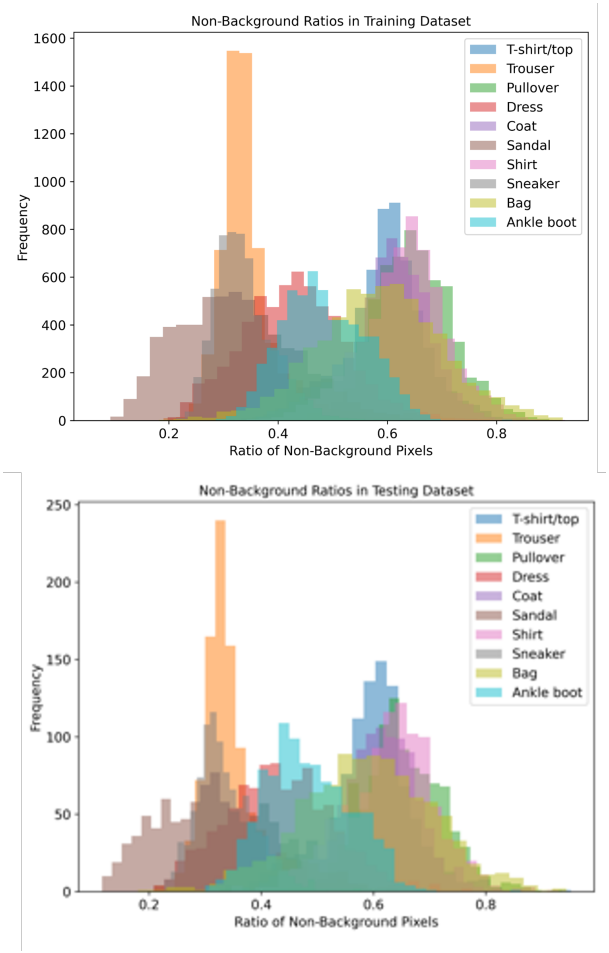


Fig. 2. Non-background ratios in Fashion-MNIST training and testing sets

Projection (UMAP) technique [7] to qualitatively visualize the intra-class feature similarities within the training set. As depicted in Figure 3, this method maps the dataset features onto a reduced dimensional space, where the proximity between points indicates the degree of similarity between different classes. For instance, we observe notable similarities between classes such as 'sneaker' and 'sandal', as well as between 'dress' and 't-shirt/top', and 'coat' and 'pullover', revealing similar patterns and relationships within the data.

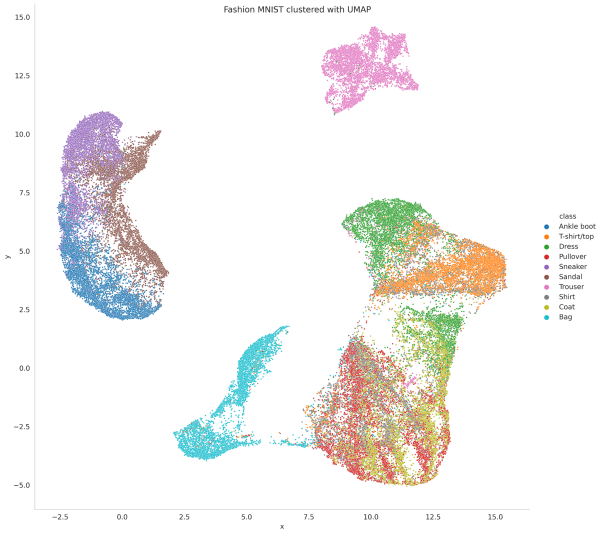


Fig. 3. Dimension reduction using UMAP on Fashion MNIST

III. METHODS

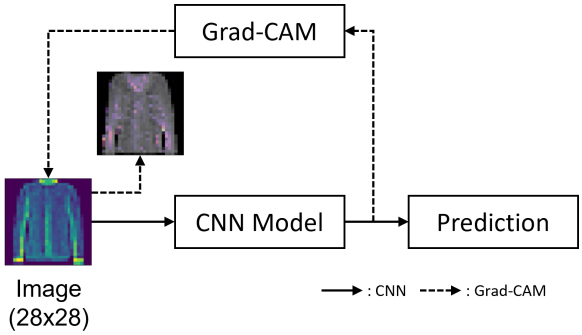
A. Algorithm selection

In our project, we have chosen a CNN for image classification and Grad-CAM as our explainable artificial intelligence (XAI) tool. The rationales behind this selection are threefold. Firstly, CNNs are widely recognized for their extensive use in a variety of computer vision applications [8]. Despite the rising prominence of transformers [9], CNNs are still well-suited for simpler datasets like Fashion-MNIST, which is less complex than datasets, such as the Microsoft common objects in context (MS-COCO) dataset [10]. This is evidenced by numerous state-of-the-art performances by CNNs on the Fashion-MNIST [11]. Secondly, we leveraged a Grad-CAM method to better understand CNN's decision-making process. This method has been extensively used as an ex-post hoc explainable technique associated with CNN-based image classifiers, allowing us to intuitively interpret its performance as a XAI tool. Lastly, this model serves as a valuable educational material, allowing us to apply and expand upon the knowledge acquired in our course. Furthermore, integrating XAI through Grad-CAM deepens our comprehension of such network, bridging theoretical learning

with our hands-on experience.

B. Model building and training

Google Colab is used for our hosted coding environment due to its comprehensive provision of essential libraries of CNN applications, ensuring the seamless execution of code for various users. Additionally, the platform's adherence to the Jupyter Notebook format ensures well-structured code presentation, enriched with markdown comments and visual aids that enhance code readability. In alignment with one of the central objectives of our project, which aims to bridge the gap between theoretical learning and practical experience, we have made the decision to construct the CNN architecture from the ground up, alongside the Grad-CAM model using Tensorflow (version: 2.14.0). Leveraging the wealth of prior knowledge applied to Fashion-MNIST, we build upon this domain-specific expertise for both architectural design and hyperparameter tuning (batch size: 64; epochs: 50; while other hyperparameters used by the default values of Tensorflow, drawing insights from references [11]–[13] for CNN and [14] for Grad-CAM. Figure 4 illustrates the schematic flow of the Grad-CAM methodology, while also providing an overview of the CNN model's architecture. For the compilation of the CNN model, we employ the Adam optimizer [15], with a focus on optimizing accuracy and minimizing categorical cross-entropy loss, as recommended in the references. Regarding the validation dataset, we perform a random split of the Fashion-MNIST training dataset, resulting in a division of 55,000 training samples, 5,000 samples for validation, and 10,000 samples for testing. Figure 5 visually demonstrates that the CNN training model exhibits no significant improvement beyond the 39th epoch, which becomes our final model for evaluation.



Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 28, 28, 64)	320
max_pooling2d (MaxPooling2D)	(None, 14, 14, 64)	0
dropout (Dropout)	(None, 14, 14, 64)	0
max_pooling2d_1 (MaxPooling2D)	(None, 7, 7, 64)	0
dropout_1 (Dropout)	(None, 7, 7, 64)	0
flatten (Flatten)	(None, 3136)	0
dense (Dense)	(None, 256)	803072
dropout_2 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 10)	2570

Total params: 805962 (3.07 MB)
Trainable params: 805962 (3.07 MB)
Non-trainable params: 0 (0.00 Byte)

Fig. 4. Schematic flow of model and model architecture of CNN

IV. Results and Discussion

A. Model evaluation

In assessing the performance of our classifier, we primarily focused on accuracy as the metric of choice. This decision is rooted in the classifier's core objective: to precisely identify types of clothing within 28x28 pixel low-resolution images, each containing a single clothing item. Our best model achieves an overall accuracy of 91.2%, demonstrating its relative competitiveness in comparison to existing benchmarks [11]. Figure 6(a) showcases the testing results from the selection of 50 randomly chosen test dataset images, highlighting only four instances of misclassification. In Figure 6(b), per-class accuracy is represented, with 'trouser,' 'sandal,' 'sneaker,' and 'bag' achieving 98% accuracy, while 'shirt' shows relatively low performance at 73%. Notably, the confusion matrix in Figure 6(c) indicates that such misclassifications on 'shirt' stem from the challenge of distinguishing features among other similar classes, such as 't-shirt/top,' 'pullover,' and 'coat.' To delve deeper into this issue, we employ the XAI tool, Grad-CAM, in the subsequent section to gain further insights.

Figure 7 offers valuable insights into our model's image classification process with the assistance of Grad-CAM. Specifically, it reveals how the model distinguishes between classes. For example, in the case of 'sandal', the model is more prone to classify it correctly when it can differentiate the instance from its background compared to the other similar classes, such as 'sneaker' and 'ankle boot'. Conversely, when the model identifies background details for 'sneaker' and 'ankle boot', it tends to misclassify them as 'sandal'. Furthermore, when the model recognizes image features along the bottom edge of an instance, it improves the accuracy of classifying 'ankle boot'. A similar pattern emerges among upper clothing classes, including 't-shirt/top', 'pullover', 'coat', and 'shirt'. Notably, the model's success in classifying 't-shirt/top' and 'pullover' relies on its focus on general image features, whereas 'coat' and 'shirt' focus on specific patterns within the instances. These interpretations provide valuable insights for understanding our model's mechanism and identifying potential areas for improvement in future model development.

We further investigate the performance of 'shirt', which exhibits the lowest accuracy among all classes, using Grad-CAM. In successful cases, the model's ability to correctly classify 'shirt' is closely related to its capacity to discern image features and patterns within the instances. This examination also highlights the inherent challenges of 'shirt' classification, primarily stemming from its visual similarities

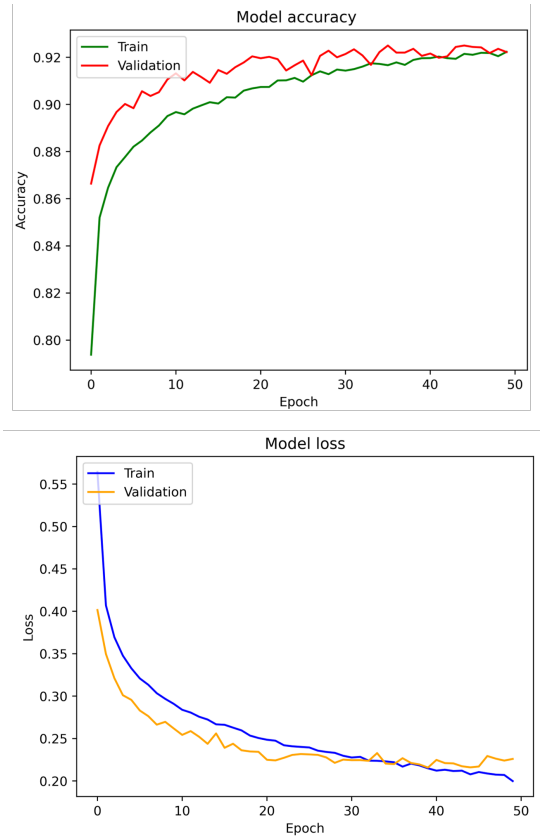


Fig. 5. CNN model accuracy and loss for train and validation datasets

with other classes. As shown in Figure 8, even human vision and intelligence can struggle with distinguishing 'shirt' from similar classes in certain cases. This insight prompts further consideration of potential improvements, particularly the necessity for higher-resolution images. Given that successful 'shirt' classification hinges on recognizing image features within instances, the improvement of higher resolution 'shirt' instances holds promise for enhanced future classification accuracy.

B. Practical implications for a clothing company

This classifier offers a promising solution for clothing factories, streamlining their parceling process through automatic filtering of clothing selections. Its application in large-scale operations, where manual classification and storage of clothing stocks are labor-intensive, not only boosts productivity but also reduces the time and effort involved for clothing companies. Moreover, leveraging advanced deep learning techniques, the system enhances image detection, segmentation, and even the creation of clothing designs through Generative Adversarial Networks (GANs). As illustrated in Figure 9 [16], the AI's impact on the fashion industry is multifaceted, encompassing fashion detection, synthesis, and personalized recommendations. To sum up, the practical benefits of this model are twofold: it is beneficial to augment the precision and automation of the classification process. Secondly, this approach

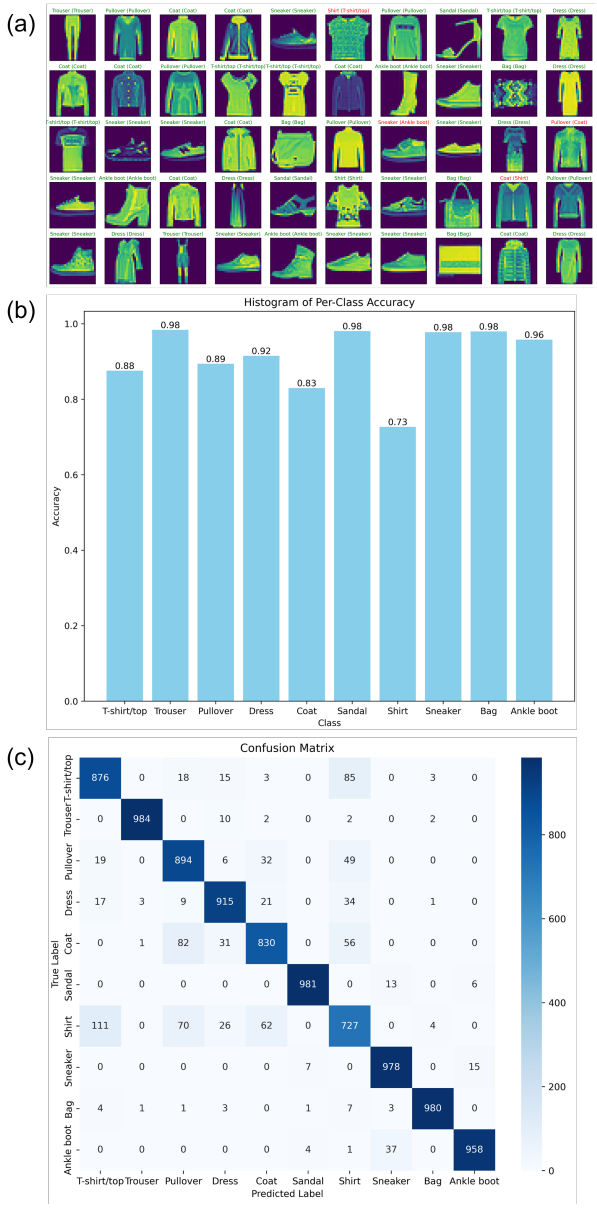


Fig. 6. Results of performance evaluation: (a) examples of results, (b) histogram of per-class accuracy, and (c) confusion matrix validation datasets

leads to AI-driven innovations, reshaping various aspects of the fashion industry. This includes generative design, user-centric virtual experiences, and cloth classification, detection, and segmentation, which are integral to developing business-centric materials in the fashion domain.

V. Conclusion

This project conducts a comprehensive analysis of the Fashion-MNIST dataset, develops and evaluates a CNN image classifier, and explores its application in the fashion industry, particularly emphasizing the classifier's performance using Grad-CAM. A notable finding is the identification of imbalances in the 'coat' class across training and testing sets, with UMAP employed to illustrate feature similarities.

Fig. 7. Examples of Grad-CAM for corrected and misclassified results per class

Fig. 8. Examples of Grad-CAM for corrected and misclassified results for shirt images

The CNN model demonstrates a commendable accuracy of 91.2% on the test set, and its decision-making processes are elucidated using Grad-CAM, offering valuable insights into the model's functionality. The project also acknowledges critical limitations. The Fashion-MNIST dataset, primarily a benchmarking tool, lacks the complexity and diversity of real-world fashion datasets. This limitation raises concerns about the model's applicability to more intricate datasets. Furthermore, the dataset's limited resolution (28x28 pixels) hinders the model's ability to capture fine details crucial for distinguishing subtle variations in fashion items. Future research should focus on applying the model to more varied and complex datasets with higher resolution images, enhancing CNN interpretability through advanced XAI techniques, and addressing class imbalances with strategies like synthetic data generation or advanced sampling. These efforts are aimed at aligning the model's capabilities more closely with the intricate requirements of the fashion industry.

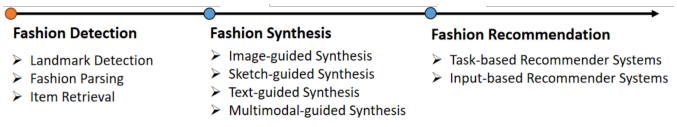


Fig. 9. Examples of benefits of AI in fashion industry

VI. Appendix

All the codes and relevant materials are shared: https://github.com/yoojunT/ECEN758_GroupAssignment

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, Art. no. 6, May 2017, doi: 10.1145/3065386.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, 2015. Available: Link

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2015, doi: 10.48550/ARXIV.1512.03385.

[4] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA: IEEE, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[6] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," 2016, doi: 10.48550/ARXIV.1610.02391.

[7] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," 2018, doi: 10.48550/ARXIV.1802.03426.

[8] A. Bakhshi, S. Chalup, and N. Noman, "Fast Evolution of CNN Architecture for Image Classification," in *Deep Neural Evolution*, H. Iba and N. Noman, Eds., in Natural Computing Series., Singapore: Springer Singapore, 2020, pp. 209–229. doi: 10.1007/978-981-15-3685-4-8.

[9] X. Huang, M. Dong, J. Li, and X. Guo, "A 3-D-Swin Transformer-Based Hierarchical Contrastive Learning Method for Hyperspectral Image Classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 60, pp. 1–15, 2022, doi: 10.1109/TGRS.2022.3202036.

[10] T.-Y. Lin, M. Maire, S.J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, "Microsoft COCO: Common Objects in Context," presented at the European Conference on Computer Vision, Zurich, Switzerland, 2014, pp. 740–755.

[11] "Image Classification on Fashion-MNIST," *Image Classification on Fashion-MNIST*. Accessed: Nov. 14, 2023. Available: Link

[12] "classification," *classification*. Accessed: Nov. 16, 2023. Available: Link

[13] "Deep Learning's Hello World, Fashion-MNIST." Accessed: Nov. 16, 2023. [Online]. Available: Link

[14] "GRAD-CAM," *GRAD-CAM*. Accessed: Nov. 16, 2023. [Online]. Available: Link

[15] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," 2014, doi: 10.48550/ARXIV.1412.6980.

[16] N. Kato, H. Osone, K. Oomori, C. W. Ooi, and Y. Ochiai, "GANs-based Clothes Design: Pattern Maker Is All You Need to Design Clothing," in *Proceedings of the 10th Augmented Human International Conference 2019*, Reims France: ACM, Mar. 2019, pp. 1–7. doi: 10.1145/3311823.3311863.