ChromaFashio[net]te: Designing Efficient Image-to-Image Translation Artificial Neural Network Model For Segmenting Fashion Images

Ahmet H. Güzel* Peihua Lai Stephen Westland

University of Leeds, Leeds, UK

Abstract

This paper proposes an approach to fashion image segmentation aiming to segment multiple classes that utilize a modified U-Net architecture and a modified Fully Convolutional Network (FCN) architecture with a weighted cross-entropy loss scheme to improve segmentation accuracy while reducing computational requirements. The proposed models were evaluated on a dataset of 2000 preprocessed fashion images with pixel-level annotations. We compared the performance of their approach against image segmentation benchmark models such as the DeepLabV3+ [26], LR-ASPP [16] and the Pix2Pix [19]. The detailed evaluation study demonstrated that the proposed approach attained comparable or higher pixel accuracy, mean pixel accuracy, and mean Intersection over Union (MIoU) scores than the benchmark models. Moreover, the proposed approach outperformed the benchmark models in terms of training and inference speeds.

1 Introduction

Color forecasting predicts consumer demand for colors in manufacturing, which traditionally takes months to complete. Color forecasting is a multi-billion pound business that is vital to many manufacturing industries [1]. With technological advancements, the time between design and retail has been reduced to as little as 18 weeks. This puts pressure on traditional expert-driven color forecasting methods, which are often inaccurate and lead to waste. Due to a large number of images, it is not practical for humans to process them manually. Therefore, over the past decade, there has been growing interest in the use of machine learning algorithms. While simple clustering techniques like k-means have been used in the past [8, 41], more advanced machine learning methods are becoming more common [13, 23, 29]. Image-based data-driven approaches using machine learning algorithms are becoming popular, but extracting dominant colors from images can be challenging. To address this, we propose using semantic segmentation to segment images based on content and analyze individual object classes for more precise color analysis. By segmenting the fashion image based on content, the object class of each pixel can be determined, allowing for more precise color analysis. This approach has the potential to provide more accurate color

forecasting by identifying which colors are associated with which objects in the image, and effectively excluding background pixels. Fashion image segmentation presents unique challenges compared to generic image segmentation, as noted by Wei et al. [20]. Firstly, clothing items on models are often subject to non-rigid deformation, resulting in variations in shape and appearance. Furthermore, there can be significant variation in the appearance of clothing items within the same class, including differences in spatial layout, style, texture, and material. All pixels are labeled with different class categories, resulting in a semantic analysis that identifies the clothing items present, their location within the image, and their respective shapes. The shape of clothing items is a crucial factor in fashion analysis as it can vary significantly, such as with different hat styles that may go in and out of fashion trends over time. Figure 1 shows the example from our dataset for the input image and the prepared segmented image by the designer.

Fig. 1. (A) Input image, (B) Segmented image

2 Related Work

2.1 Generic Semantic Segmentation

Semantic segmentation, which involves pixel-level classification of an image into different object or scene categories, has been an active research area in computer vision. With the advent of deep learning, CNNs have become the dominant approach for semantic segmentation. Fully Convolutional Networks (FCNs) [26]

and U-Net [32] are popular architectures that extend CNNs to perform pixel-wise predictions, producing accurate and detailed semantic segmentation maps. A conditional Generative Adversarial Network (GAN) based model was proposed for image-to-image translation, including semantic segmentation task [19]. Other variants, such as DeepLab [5] and Gated-Shape CNN [35], have also been proposed to improve the performance of semantic segmentation using different strategies, such as atrous convolutions and two-stream approach, respectively. Our study drew inspiration from CNN-based models and conducted a comparative analysis with our proposed model regarding computational cost and the accuracy of the targeted dataset.

2.2 Fashion Image Segmentation

In fashion image segmentation, which encompasses clothing retrieval and parsing, early approaches predominantly utilized handcrafted features like SIFT [27], HOG [9], and color histograms. However, the performance of these methods was hindered by their limited ability to capture the nuanced characteristics of clothing items accurately [3, 4, 11, 37]. Additionally, there has been research conducted by multiple scholars in the field of clothing parsing, utilizing sophisticated computer vision algorithms such as super-pixels and human pose detection [24, 30], hierarchical image segmentation [38], and conditional random field [10]. Recently, machine learning models have been developed to acquire more discriminative representations that can effectively tackle variations across different scenarios [7,17,25,28,36]. Most of these studies target semantic segmentation of clothing items, as it has potential applications in the fashion industry, such as fashion trend prediction and image-based information retrieval. In our work, we proposed and aimed to create a segmentation pipeline that can be used as a part of an efficient and fast end-to-end process for color forecasting approaches in the fashion industry. To achieve this, we investigated the use of Artificial Neural Network (ANN) based models for segmenting clothes, accessories, human skin, and hair to provide accurate and efficient semantic information to improve color forecasting approaches.

3 Problem Formulation

By segmenting the image based on content, the object class of each pixel can be determined, allowing for more precise color analysis. Successful semantic segmentation also allows for the effective exclusion of background pixels. Fashion image segmentation is a task of pixel-level classification that aims to segment fashion-related objects, such as clothing items, accessories, human skin, and hair, from images. The goal is to accurately delineate the boundaries of fashion objects and assign them semantic labels, such as dresses, sunglasses, necklaces, shoes, etc. This task is challenging due to the variability in fashion styles, poses, lighting conditions, occlusions, and complex object interactions. Formally, given an input image I of size $W \times H$, the goal of fashion image segmentation is to produce a

segmentation map S of the same size, where each pixel S_{ij} is assigned a semantic label from a predefined set of fashion object categories. The segmentation map should accurately capture the object boundaries and provide fine-grained semantic labels for each pixel. Mathematically, the problem can be formulated as:

$$S = F(I) \tag{1}$$

where F is the ANN-based segmentation model that takes an input image I and produces the corresponding segmentation map S. Moreover, this research aims to develop an alternative solution to fashion image segmentation that is faster and more accurate than the current Pix2Pix [19] model used by the Color Science group at the University of Leeds. While Pix2Pix has effectively produced high-quality image segmentation, its computational requirements are high, making it impractical for some applications. Additionally, Pix2Pix can be challenging to train and requires significant data pre-processing and post-processing. To address these challenges, we propose a new approach to fashion image segmentation that is both faster and more accurate. Our approach utilizes a combination of preprocessing, improved multi-class classification loss, and optimized CNN architecture to process image data, reducing the computational requirements while maintaining or improving segmentation accuracy. To evaluate the effectiveness of our approach, we will compare it against the current state-of-the-art models including Pix2Pix on the dataset, using both pixel-based classification and similarity measures.

4 Method

This section provides an overview of the data collection and ground truth masking process, which involves creating pixel-level masks for our semantic segmentation problem. We then describe the benchmark and proposed CNN models, including their testing, training process, and hyperparameter optimization details. Finally, we elaborate on the evaluation metrics used to compare the performance of the deep learning models for our specific problem.

4.1 Data Pre-Processing

To prepare the dataset for training models, several steps were performed. Firstly, 2000 images with the "streetstyle" tag were collected from Google. Next, data cleansing was applied during the preparation process, and all images in the annotation task were resized to the same dimensions of 512 x 256 pixels. To perform the pixel-level semantic segmentation task, designers were hired, and annotations were completed using Photoshop. For each original image, an annotated image was generated, where each pixel was assigned one of five colors corresponding to one of the five classes: skin, hair, accessories (e.g., hat, shoes, and bag), clothes, and background (null). Each input image was associated with a corresponding pixel-level annotated image, and an example of this can be seen in Figure 2, which shows an input image and its segmented ground truth version with given

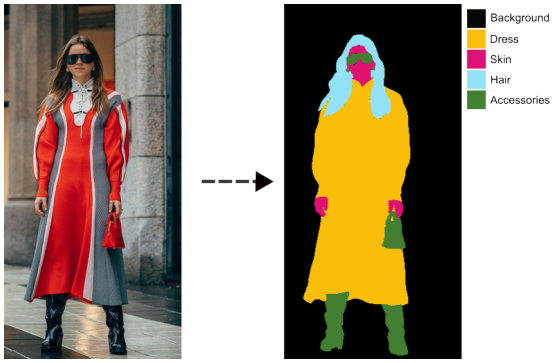

Fig. 2. The input image is converted to a ground truth image with five different masks as background, dress, skin, hair, and accessories.

color channel labels. The pixel-wise cross-entropy loss is image segmentation's most widely used loss function [5,26,35]. This loss evaluates each pixel separately by comparing the predicted class (a pixel vector) to the one-hot encoded target vector. To create target labels suitable for cross-entropy loss, we developed an algorithm to classify each pixel regarding its color from the ground truth (prepared) images. Firstly, Red, Green, and Blue (RGB) channels are converted to greyscale to avoid additional computational costs during the data loading before input to the CNN model. Then, each greyscaled image from the ground truth-masked images is converted to class labels for each pixel. Figure 3 represents classes for each color channel. For the experiments conducted in this study, the dataset was randomly split into training and testing sets using an 80-20 ratio, where 80% of the data was used for training the model, and 20% of the data was reserved for evaluating the model's performance. Additionally, normalization is investigated as a data augmentation method to investigate the performance of the model's segmentation task.

Network Design and Training

This section describes our study's semantic segmentation models: FCN, U-Net. Our work uses per-pixel classification, applying a classification loss to each output pixel [7].

U-Net U-Net [32] is a popular semantic segmentation architecture known for its U-shaped encoder-decoder structure. U-Net consists of an encoder network that captures contextual information from the input image, followed by a decoder network that generates segmentation maps with skip connections that concatenate features from the encoder to the corresponding decoder layers. Both encoder and decoder use the form 2D convolution modules, 2D batch normalization, and rectified linear unit activation function [18]. U-net architecture with input/output

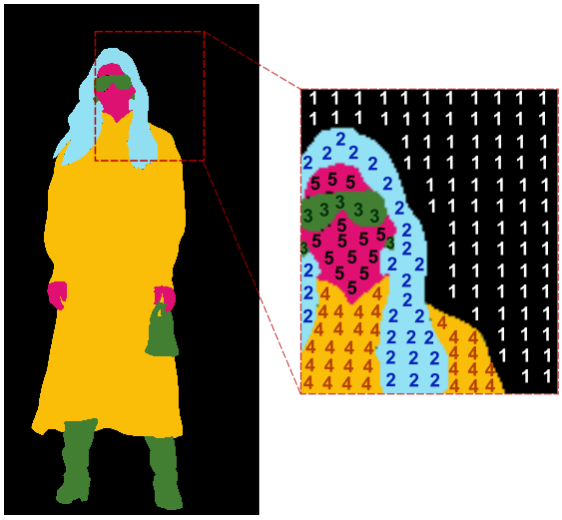

Fig. 3. The unique color channels are assigned to class labels for each pixel-wise to create labels for cross-entropy loss.

pipeline is diagrammed in Figure 4. The input image has three channel image tensors with selected batch size, and the output from the U-Net is the same batch size with five single channel tensors with predicted classes for each pixel. The softmax function is applied to the output tensor to obtain predicted classes. The segmented image prediction is then decoded based on the ground truth colors, using the output of the softmax function, which provides the highest probability classes for each pixel. We designed different U-Net architectures by varying the depth (downsamplings, upsamplings) between 4 and 7, and the channel dimensions between 64 and 192 to experiment with segmenting performance and computational cost.

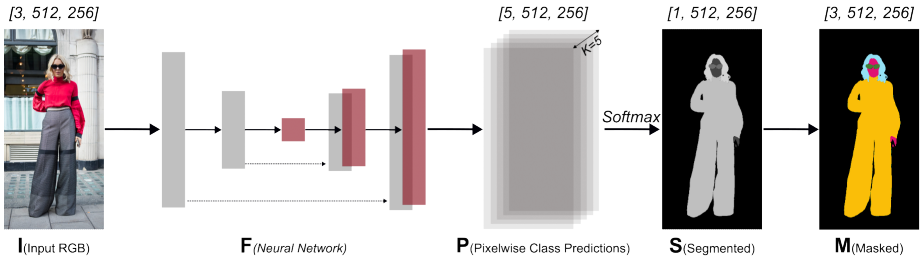

Fig. 4. The "U-Net" pipeline is characterized by its encoder-decoder structure, where the encoder and decoder stacks have mirrored layers, and skip connections are established between them.

Fully Connected Network Another CNN architecture that inspired us to design our model is the Fully Connected Network (FCN). FCN model preserves the spatial information from lower-resolution feature maps and combines them with higher-resolution feature maps to improve the segmentation accuracy [26]. FCN model can be constructed with a backbone CNN which can be AlexNet [22], GoogLeNeT [34], ResNet [15], and VGG16 [33]. In our research, we chose to use the FCN with VGG16 backbone because prior studies have demonstrated that VGG performs exceptionally well regarding pixel accuracy and mean intersection union (IoU) [26]. Our FCN architecture with VGG16 backbone with input/output pipeline is diagrammed in Figure 5. Initially, we selected FCN32s, FCN16s, and FCN8s with final upsampling rates of 32, 16, and 8, respectively, as used in the earlier work [26]. We then improved these models by modifying their final layer with an upsampling ratio of 1 as FCN1s.

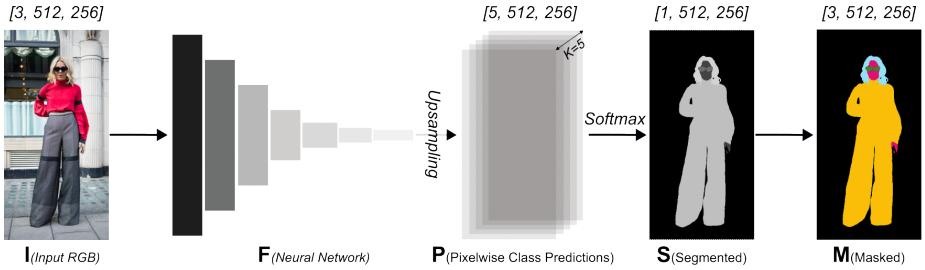

Fig. 5. The "FCN" pipeline is using pre-trained VGG16 backbone with a final upsampling layer with different upsampling rates.

Benchmark Models To evaluate our models against benchmark models in the literature, we chose recent and complex models: DeepLabv3+ [6], and Pix2Pix [19], which are popular deep learning models for semantic segmentation tasks. DeepLabv3+ uses an encoder-decoder architecture with Atrous Spatial Pyramid Pooling (ASPP), a semantic segmentation module for resampling a given feature layer at multiple rates prior to convolution to capture context and handle different scales. On the other hand, Pix2Pix is a conditional Generative Adversarial Network (cGAN) that generates realistic segmentation maps from input images. Moreover, we use LR-ASPP [16] as another benchmark network capturing contextual information at multiple scales by applying atrous convolution at different dilation rates to the feature maps.

4.2 Model Training

Loss Model Our work uses pixel-wise cross-entropy loss, which compares class predictions to target vectors on a per-pixel basis. Since each pixel is treated

equally during loss calculation, the loss may result in issues with imbalanced class representations. To tackle this, Long et al. [26], in their FCN paper, propose weighting the loss for each output channel, accounting for class imbalances in the dataset. This approach aims to give more importance to minority classes during training, mitigating the impact of imbalanced class distributions. To demonstrate the effectiveness of this approach, we evaluated our models both vanilla cross-entropy loss and weighted cross-entropy loss function that is applied dynamically in the training loop for each batch of data. By dynamically adjusting the loss function for each batch, we aimed to remove the negative effects of imbalance in the data and improve segmentation accuracy. In our fashion image dataset, cross-entropy loss weights for each class are calculated over all training image dataset by using the following Equation 2

$$\mathbf{w} = \frac{1 - \rho_i}{N - 1}, \tag{2}$$

where $\mathbf{w}$ represents the weight vector for 5 different classes, ρ_i represents the ratio of the number of pixels to total pixels in the training batch for each class, and N is the number of classes. Then, we use $\mathbf{w}$ to construct weighted cross entropy loss calculation during the training. We also used a vanilla version of cross entropy loss treating each pixel equally during the loss calculation step. Equation 3 is to formulate our loss model for pixel-wise semantic segmentation,

$$CrossEntropy(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} w_c \cdot y_{i,c} \cdot \log(\hat{y}_{i,c}) \tag{3}$$

where: y represents the ground truth labels for the image, $y_{i,c}$ denotes the ground truth label for pixel i in class c, $\hat{y}$ represents the predicted class probabilities, $\hat{y}_{i,c}$ denotes the predicted probability of pixel i belonging to class c, N represents the total number of pixels in the image, C represents the number of classes in the segmentation task, and w_c represents the weight assigned to each class c to account for class imbalance. For the pixel-wise vanilla multi-class cross-entropy loss, all weights are set to 0.2 for the five classes segmentation task.

Optimization For the training step, we use PyTorch deep learning library's [31] Adam optimizer which combines elements of momentum and RMSprop, and it adjusts the learning rate based on the mean and variance of the gradients, allowing for faster convergence and improved performance in training deep neural networks [21]. Our preferred Adam optimizer setting in Pytorch is betas (0.9, 0.999), epsilon 1e-08, and weight decay 0. We tried different learning rates during the hyper-parameter optimization stage and found that the best results are obtained using a learning rate is 1e-4. Our hardware for training models is NVIDIA RTX 3070Ti GPU, which allows a maximum of 8 batches of images during each training. Moreover, we found that the number of epochs is needed to be set differently for proposed CNN architectures. We selected the number of epochs for the U-Net-based architectures as 15; for FCN, we used 10 epochs to converge the

minimum loss value. To improve the speed of the training, we also investigate mixed precision training enabled by NVIDIA CUDA cores in our GPU. Mixed precision training automatically selects the 32-bit floating point tensor type to be replaced with a 16-bit floating point during training without affecting the accuracy of training. We found that mixed precision training improves classical training by 1.6 times in terms of speed with less than a 0.5 percent change in accuracy.

4.3 Evaluation

Metrics We utilize two metrics commonly used in semantic segmentation and scene parsing evaluations in the testing dataset: pixel accuracy and Mean Intersection over Union (MIoU) Let n_{ii} represents the number of correctly predicted pixels for class i and where there are N_{cl} different classes, and let n_{ij} is the number of pixels of class i predicted to belong to class j, where t_i the total number of pixels in the ground truth region for class i, we compute following metrics for our evaluation:

$$PixelAccuracy = \frac{\sum_i n_{ii}}{\sum_i t_i} \tag{4}$$

$$MeanPixelAccuracy = \frac{1}{N_{cl}} \frac{\sum_i n_{ii}}{\sum_i t_i} \tag{5}$$

$$mIoU = \frac{1}{N_{cl}} \sum_i \frac{n_{ii}}{t_i + \sum_j n_{ji} - n_{ii}} \tag{6}$$

Additionally, we use MIoU without background class since the class is dominant over other classes regarding the number of pixels in each image.

Training/Inferring speed To evaluate the training and inference speed of each model with a fixed batch size of 8, we utilize asynchronous operation between CPU and GPU when the model is loaded to GPU. To measure the inference speed accurately, we address two caveats by running dummy examples for GPU warm-up to prevent power-saving mode, and using *torch.cuda.synchronize()* to ensure synchronization between GPU and CPU during time measurements [12]. This approach overcomes the issue of unsynchronized execution, ensuring reliable and accurate time recordings for evaluating the speed of our approach. The training loop pseudo-code for our approach is illustrated in Algorithm 1.

4.4 Results

U-Net Table 1 presents the performance results of U-net based architectures using selected evaluation metrics, including pixel accuracy (Eq. 4), mean pixel accuracy (Eq. 5), and mIoU (Eq. 6).

Algorithm 1 Training Loop for Weighted Cross Entropy Loss

```
for epoch in range(num_epochs) do
    for data in training set do
        inputs, targets ← data
        w ← calculate_class_weights(targets)
        loss_function ← WeightedCrossEntropyLoss(w)
        optimizer.zero_grad()
        predictions ← cnn_model(inputs)
        loss ← loss_func(predictions, targets)
        loss.backward()
        optimizer.step()
    end for
end for
```

Table 1. Pixel accuracy results for U-Net architectures on test data using vanilla cross-entropy loss.

Model	Dims.	Layer	Null	Hair	Clothes	Skin	Accs.	Mean	MIoU
U-Net1	64	4	0.972	0.569	0.848	0.755	0.37	0.924	0.53
U-Net2	64	5	0.976	0.509	0.873	0.753	0.37	0.925	0.515
U-Net3	64	6	0.983	0.474	0.892	0.802	0.422	0.936	0.534
U-Net4	64	7	0.988	0.636	0.904	0.763	0.372	0.941	0.567
U-Net5	128	7	0.99	0.694	0.903	0.81	0.414	0.947	0.604
U-Net6	192	7	**0.992**	**0.741**	**0.908**	**0.845**	**0.520**	**0.951**	**0.642**

The results show that increasing the depth and the dimension of a layer improves pixel accuracy, mean pixel accuracy, and MIoU. Due to GPU limitations, the experiments were conducted up to a dimension of 192 and a layer of 7 for the U-net. In our evaluation, we chose the U-Net6 for comparison against other architectures and benchmark models utilized.

FCN Table 2 presents the performance results of FCN based architectures using selected evaluation metrics, including pixel accuracy (Eq. 4), mean pixel accuracy (Eq. 5), and mIoU (Eq. 6). The results show that reducing the final up-sampling improves pixel accuracy, mean pixel accuracy, and MIoU. We select the FCN1s to compare other architectures and benchmark models we used in the evaluation.

Benchmark Models In this section, we provide the best U-net and FCN architecture to compare with DeeplabV3+, LR-ASPP, and Pix2Pix. Table 3 summarizes four different models' performance in terms of metrics, and the combined training and inferencing speed. Table 3 shows that FCN1s is the best custom design model against benchmark models. Among the compared networks, FCN1s demonstrate superior pixel accuracy for each class, with the exception of

Table 2. Pixel accuracy results for FCN architectures on test data using vanilla cross-entropy loss.

Model	Upsample	Null	Hair	Clothes	Skin	Accs.	Mean	MIoU
FCN-32s	32x	0.988	0.764	0.939	0.781	0.571	0.958	0.654
FCN-16s	16x	0.991	0.822	0.944	0.853	0.579	0.964	0.698
FCN-8s	8x	0.990	0.793	0.904	0.925	0.649	0.960	0.685
FCN-1s	1x	**0.992**	**0.835**	**0.955**	**0.867**	**0.607**	**0.966**	**0.715**

Table 3. Pixel accuracy results for U-Net/FCN/Benchmark architectures on test data using vanilla cross-entropy loss.

Model	Null	Hair	Clothes	Skin	Accs.	Mean	w/o Null	MIoU	Speed
Pix2Pix	0.973	0.772	0.928	0.805	0.402	0.926	0.815	0.640	1.0x
LR-ASPP	0.989	0.756	0.934	0.815	0.569	0.959	0.832	0.657	5.4x
DeeplabV3+	**0.994**	0.815	**0.964**	**0.884**	0.585	**0.971**	**0.880**	**0.732**	1.9x
U-Net6	0.988	0.741	0.908	0.845	0.520	0.951	0.801	0.642	2.1x
FCN1s	0.992	**0.835**	0.955	0.867	**0.607**	0.966	0.862	0.715	3.6x

DeeplabV3+. Notably, FCN1s outperforms DeeplabV3+ specifically in classifying hair and accessories. These findings suggest that FCN1s is particularly effective in accurately classifying thin shape classes in comparison to DeeplabV3+. In addition, FCN-1s exhibits a considerable advantage in terms of processing speed, with an almost twofold increase in speed compared to DeeplabV3+. This noteworthy performance gain makes FCN1s a more efficient choice for real-time or time-sensitive applications, where faster processing speeds are crucial for practical deployment.

Weighted Cross-Entropy Loss We also tested the weighted cross-entropy-based performance of designed networks to evaluate the positive effects of the weighted loss approach. The results presented in Tables 4 and 5 show that us-

Table 4. Pixel accuracy results for FCN1s architecture on test data, comparing vanilla cross-entropy loss and weighted cross-entropy loss.

Model	Null	Hair	Clothes	Skin	Accs.	Mean	w/o Null	MIoU
FCN1s	0.992	0.835	0.955	0.867	0.607	0.966	0.862	0.715
FCN1s-**wl**	0.990	0.840	0.955	0.903	0.638	0.968	0.871	0.732

ing weighted cross-entropy loss leads to better performance compared to vanilla cross-entropy loss. This improvement can be attributed to the weighting approach, which assigns higher importance to underrepresented classes such as hair, skin, and accessories. By giving these classes more weight during training, the model is able to better learn their characteristics, leading to enhanced segmentation performance. At the same time, the weighting approach does not negatively impact the performance of well-represented classes like null and clothes, which maintain their performance levels from the vanilla approach. This indicates that the weighted cross-entropy loss strikes a balance between focusing on underrepresented classes while preserving the performance of the more common classes.

Table 5. Pixel accuracy results for DeeplabV3+ benchmark model on test data, comparing vanilla cross-entropy loss and weighted cross-entropy loss.

Model	Null	Hair	Clothes	Skin	Accs.	Mean	w/o Null	MIoU
DeeplabV3+	0.994	0.815	0.964	0.884	0.585	0.971	0.880	0.732
DeeplabV3+-**wl**	0.989	0.870	0.964	0.892	0.693	0.970	0.884	0.741

5 Discussion

5.1 Limitations

While the proposed approaches show promise for fashion image segmentation, there are some limitations to the study that should be noted. One limitation is that the dataset used in the study is relatively small, consisting of only 2000 images. This limited dataset may not be representative of the full range of fashion styles and poses, which could impact the generalizability of the proposed approach. Additionally, despite our proposed weighted cross-entropy loss, the high degree of class imbalance in the dataset remains a limitation. Specifically, the number of pixels for hair, skin, and accessories is less than a magnitude of 10 compared to the other classes. This class imbalance can impact the training of the segmentation model, resulting in a suboptimal performance for these classes.

5.2 Future Work

One potential area for future research is to investigate different network models that could improve the accuracy and efficiency of fashion image segmentation. While we utilized U-Net and FCN models, there are several other popular network models that could be explored, such as Mask R-CNN [14], PSPNet [40]. Moreover, there may be more opportunities to optimize the hyper-parameters of

the existing network models to improve their performance via exploring advanced hyper-parameter optimization methods, and recently introduced optimizers like automatic gradient descent [2]. Improving semantic segmentation accuracy by addressing the class imbalance issue is a significant area for future research. This is because underrepresented classes, such as accessories (e.g., wristlets, anklets, necklaces, etc.), can also be complex in terms of shape. We tried augmenting the training data by normalizing the input images, but this yielded no noticeable improvement. However, there are other advanced augmentation methods that could have the potential for further improvement [39].

6 Conclusion

The paper focuses on the task of fashion image segmentation, which is a challenging problem due to the variability in fashion styles, poses, lighting conditions, and complex object interactions. We propose an approach to fashion image segmentation that utilizes a combination of preprocessing and CNN to reduce computational requirements while maintaining or improving segmentation accuracy. Specifically, we designed and evaluated modified versions of U-Net and FCN architectures on a dataset of 2000 images with pixel-level annotations for five different class categories. We compared the performance of their proposed approach against the DeeplabV3+, LR-ASPP, and the current Pix2Pix model used by the Color Science group at the University of Leeds on the same dataset using various evaluation metrics such as pixel accuracy, mean pixel accuracy, and mean Intersection over Union (MIoU). It is revealed that enhancing the depth and dimension of layers in the U-Net architecture leads to improvements in pixel accuracy, mean pixel accuracy, and MIoU with computational cost penalty. Furthermore, the highest performance among modified architectures was observed when the upsampling ratio was set to 1 for the FCN network. Modified FCN outperforms the benchmark models in terms of training and inference speeds, albeit at a slight expense of pixel accuracy and MIoU when compared to the best-performing benchmark model. As a recommendation for the Colour Science Group at the University of Leeds, we suggest considering the replacement of the Pix2Pix model with the modified FCN model in their color forecasting end-to-end model study. By replacing the Pix2Pix with the proposed FCN, we anticipate improvements in clothing segmentation accuracy, along with faster training and inference speeds. Moreover, the proposed model is expected to eliminate the need for post-processing edits currently needed for Pix2Pix, resulting in a more efficient and streamlined process.

7 Acknowledgment

I would like to thank Paihua Lai for their diligent work in preparing the dataset, which laid the foundation for our research. I am also deeply grateful to Prof. Stephan Westland for initiating the project, providing the initial idea, and offering insightful reviews throughout the process. Lastly, I extend my appreciation

to Dr. Muhammad Babar, and Dr. Abdulrahman Altahhan for their valuable input and careful evaluation during key meetings. Their collective support and guidance have been instrumental in the completion of this work.

References

1. E. Barnett. Trend spotting is the new £36bn growth business. `https://www.telegraph.co.uk/finance/newsbysector/mediatechnologyandtelecoms/8482964/Trend-spotting-is-the-new-36bn-growth-business.html`, 2011. Accessed: April 6, 2023.
2. J. Bernstein, C. Mingard, K. Huang, N. Azizan, and Y. Yue. Automatic Gradient Descent: Deep Learning without Hyperparameters. *arXiv:2304.05187*, 2023.
3. L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool. Apparel classification with style. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, volume 7727 of *Lecture Notes in Computer Science*, pages 4957–4968, 2012. Conference paper, 4957 Accesses, 69 Citations.
4. H. Chen, A. C. Gallagher, and B. Girod. Describing clothing by semantic attributes. In *European Conference on Computer Vision*, 2012.
5. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. 2017.
6. L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation, 2018.
7. B. Cheng, A. G. Schwing, and A. Kirillov. Per-pixel classification is not all you need for semantic segmentation. 2021.
8. Y. Dai, Y. Chen, W. Gu, Y. Tan, and X. Liu. Color identification method for fashion runway images: An experimental study. *Color Research & Application*, 2022.
9. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005.
10. W. Fan, Z. Qiyang, Y. Baolin, and X. Tao. Parsing fashion image into mid-level semantic parts based on chain-conditional random fields. *IET Image Processing*, 10(7):536–543, 2016.
11. J. Fu, J. Wang, Z. Li, M. Xu, and H. Lu. Efficient clothing retrieval with semantic-preserving visual phrases. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, volume 7725 of *Lecture Notes in Computer Science*, pages 3929–3939, 2012. Conference paper, 3929 Accesses, 22 Citations.
12. A. Geifman. The correct way to measure inference time of deep neural networks, 2020.
13. A. Han, J. Kim, and J. Ahn. Color trend analysis using machine learning with fashion collection images. *Clothing and Textiles Research Journal*, 40(4):308–324, 2022.
14. K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn, 2018.
15. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. 2015.
16. A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam. Searching for mobilenetv3. 2019.

17. J. Huang, R. S. Feris, Q. Chen, and S. Yan. Cross-domain image retrieval with a dual attribute-aware ranking network, 2015.
18. S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 2015.
19. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.
20. W. Ji, X. Li, F. Wu, Z. Pan, and Y. Zhuang. Human-centric clothing segmentation via deformable semantic locality-preserving network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(12):4837–4848, 2020.
21. D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv*, Dec. 2014.
22. A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.
23. P. Lai and S. Westland. Machine learning for colour palette extraction from fashion runway images. *International Journal of Fashion Design, Technology and Education*, 13(3):334–340, 2020.
24. S. Liu, J. Feng, C. Domokos, H. Xu, J. Huang, Z. Hu, and S. Yan. Fashion parsing with weak color-category labels. *IEEE Transactions on Multimedia*, 16(1):253–265, 2014.
25. Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1096–1104, 2016.
26. J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. 2015.
27. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 74k Accesses, 37179 Citations, 76 Altmetric Metrics.
28. J. Martinsson and O. Mogren. Semantic segmentation of fashion images using feature pyramid networks. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3133–3136, 2019.
29. K. Nadeem, M. Ahmad, Z. Javed, and M. A. Habib. Development of a machine learning model for prediction of colour trends in fashion industry. In *2022 International Conference on Frontiers of Information Technology (FIT)*, pages 296–301, 2022.
30. D. S. I. H. C. J. Nataraj Jammalamadaka (IIIT Hyderabad), Ayush Minocha (IIIT Hyderabad). Parsing clothes in unrestricted images. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013.
31. A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.
32. O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. 2015.
33. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. 2015.
34. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. 2014.
35. T. Takikawa, D. Acuna, V. Jampani, and S. Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. 2019.
36. P. Tangseng, Z. Wu, and K. Yamaguchi. Looking at outfit to parse clothing, 2017.

37. X. Wang and T. Zhang. Clothes search in consumer photos via color matching and attribute learning. In *Proceedings of the 19th ACM International Conference on Multimedia*, page 1353–1356, 2011.
38. K. Yamaguchi, M. H. Kiapour, L. E. Ortiz, and T. L. Berg. Parsing clothing in fashion photographs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3570–3577, 2012.
39. S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen. Image data augmentation for deep learning: A survey. 2022.
40. H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. 2017.
41. L. Zhao, Z. Wang, Y. Zuo, and D. Hu. Comprehensive evaluation method of ethnic costume color based on k-means clustering method. *Symmetry*, 13(10), 2021.