A Study on Fairness and Diversity in Gender Classification

Maliha T Islam
College of Information & Computer Sciences
University of Massachusetts Amherst
mtislam@cs.umass.edu

Zafeiria Moumoulidou
College of Information & Computer Sciences
University of Massachusetts Amherst
zmoumoulidou@cs.umass.edu

Abstract

In this project, we empirically evaluate how a fair and diverse dataset affects the behavior of a gender classification model that uses a simple CNN architecture. Towards this end, we train the same model on two different datasets; one biased and another one which is de-biased. We show that the overall performance of a classifier trained on a fair and diverse dataset is better even after applying data-level techniques such as random over-sampling of the minority class or SMOTE to address the presence of bias. As a second step, we experiment with ensemble models and check whether building an independent model for an under-represented group in which a classifier under performs helps in boosting the accuracy. We conclude, that ensemble models alone are not capable of mitigating the problem. Introducing fake or artificial data, even if they are generated in unusual ways, can provide support for the minority group as long as the generated data is similar.

1. Introduction

Recent studies have shown that existing software can discriminate on various sensitive attributes such as gender or race. For instance, Buolamwini et al. [2] showed that commercial gender classification systems that use image data, exhibit lower accuracy in certain demographic groups. As a result, in the recent literature there have been various approaches in order to address the presence of bias in the decision making and learning process. Nonetheless, the majority of the efforts have focused on designing robust models that do not discriminate without taking explicitly into account that the data affect the learning process. In particular, the data can be unbalanced and inherit biases so it is important that we systematically design automated processes to generate training samples over a larger dataset that fairly represent the world population. Another important aspect of the data, apart from *fairness*, is *diversity* which plays an important role in building generalizable models.

For this project, at first we empirically evaluate the effect that a *fair* and *diverse* training set has in the behavior of a CNN gender classification system. For *diversity*, we use the MaxMin diversity[8] metric defined as the minimum pairwise distance of the data points in a set. We define the *fairness* of a set with respect to a sensitive attribute such as gender, as a set of cardinality constraints over the various classes.(e.g a set has to include 20 women and 20 men so as to be as to be fair, thus demonstrate a 50-50% balance). For the empirical evaluation, we train the same model with the same hyperparameters using two different training datasets; a *fair* and *diverse* and a highly unbalanced and non-diverse one. Then, we check how the overall and the per-class accuracy of the classifier changes. Since accuracy is not sufficient in the presence of imbalances, we also compare the precision and recall scores of the two classifiers. We show that a *fair* and *diverse* training set results in a classifier with high precision and recall with good overall and per-class accuracy. On the other hand the classifier trained on a highly unbalanced dataset has high precision but low recall. Finally, we use *data-level* approaches, such as over-sampling and the more sophisticated Synthetic Minority Over-sampling Technique (SMOTE) in an effort to build a better classifier. Nonetheless, although recall gets better, precision drops. We conclude that the performance of the classifier trained on a *fair* and *diverse* set remains superior.

Apart from the *data-level* approach described above, the next thing we investigate is whether a *model-level* approach could help in building more fair and accurate classifiers across different groups. In particular, we experiment with the idea of ensemble models where each model is trained on a different sub-sample of the original data. We evaluate whether this approach boosts the accuracy of a classifier with respect to a group of interest, while maintaining the overall accuracy.

To create the sub-samples that are used to train the model ensembles, we use different sampling techniques like random and stratified sampling. The UTKFace dataset we use in the project, severely under-represents people aged over 60 years and clearly under performs when it comes to this

population. So we also explore a way of artificially generating samples for the under represented groups and evaluate the effect of training on training datasets augmented with these artificial data. The results attained from these experiments suggest that ensembles can be faster to train because of using subsets of data as opposed to using the entire dataset. However, the low performance for under-represented groups cannot be alleviated with ensemble techniques alone. From our findings we conclude that the lack of data pertaining to a certain group can be partially mitigated if methods of artificial data generation is used.

2. Related Work

Diversity is an important concept in the machine learning community and there have been recent approaches to release diverse and balanced datasets. IBM released a novel dataset, the Diversity in Faces (DiF) [9] that was created by taking into account different definitions of diversity and aims to help in creating more fair and accurate facial recognition algorithms. Another recent attempt is that of Kärkkäinen and Joo [5] who generate the FairFace dataset [5] and show that its diversity and balanced representation across different demographic groups helps in building classifiers with better generalizable performance. In particular, they perform an empirical evaluation in the performance of an identical model architecture, ResNet-34, which was trained on different publicly available datasets, including the UTKFace. Finally, in the recent work of Celis et al.[3], where they design a fair sampling algorithm using a volume-based diversity definition different that ours, they leave the empirical evaluation of the effect a *fair* and *diverse* sampler has on the accuracy of the classifier as a future direction.

From the technical perspective, several approaches have been developed in the past for the task of gender classification from images. One of the first approaches with CNNs was the architecture of Levi et al. [6] where they use a simple, shallow network and show its competence in this task. Another recent approach uses a hybrid structure of a Convolutional Neural Network (CNN) with Extreme Learning Machine (ELM) [4]. Grigory Antipov and Sid-Ahmed Berrani[1] conducted their research on gender recognition from unconstrained facial images in a cross-dataset protocol using Labeled Faces In The Wild.

Our goal here is to not only focus on gender classification, rather mitigate the discriminatory behaviour towards certain demographic groups. The Gender Shades project developed by Buolamwini et al.[2] evaluates the accuracy of AI powered gender classification methods used in practice. This specifically addresses why further work is needed to bridge the gap in accuracy for certain minorities and sensitive groups.

3. Approach

The dataset we used for the project is the UTKFace dataset [10] that consists of 23,708 RGB images of size 200×200. It is a balanced dataset with respect to gender since it contains $\sim 48\%$ female and $\sim 52\%$ male samples. However, this is not the case with respect to the age attribute. In particular, we noticed that approximately one third of the images are of people aged between 15-30 years old. The most under represented group is that of people aged over 60 (less than 3% of the dataset). Faces which are less than 10 years old are also significantly less in number (approximately 15% of the dataset). So there is a substantial amount of imbalance in the age distribution of the images.

3.1. Data-level Approach

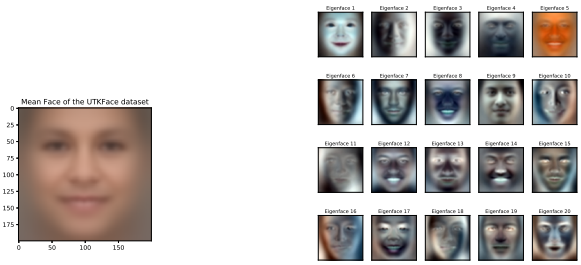
In this part of the project, we evaluate the hypothesis that a *fair* and *diverse* sample helps in building better gender classifiers. Towards this end, we train the same classifier on a *fair* and *diverse* dataset and on a second dataset that is highly unbalanced and non-diverse. Before analyzing the CNN gender classification model we used, we first explain how the two training datasets were generated.

3.1.1 Generation of datasets

We create two training datasets of size $k = 15,000$ images each so as to be able to make the comparisons. For the *fair* and *diverse* dataset, we use the Fair-and-Diverse Sampling algorithm which was designed as part of an active research project. We characterize a set as fair if the ratio of the two classes, female and male, is close to one. In the input of the algorithm we define the required cardinality per class as two integers k_1 and k_2 respectively that sum up to k. Moreover, we compute the pairwise distances of the elements in the original set and store it in a matrix which is given as an input to the algorithm.

Briefly the algorithm runs in two phases. In the first phase, we retrieve a diverse sample of size k ignoring the fairness constraints. The diverse only algorithm, it first randomly picks an elements and at each subsequent step it picks the item whose minimum pairwise distance with respect to the items already in the sample is the maximum. Then if the retrieved sample of size k does not satisfy the fairness constraints, it means that one of the classes is over represented and the other is under represented. So in the second phase of the algorithm, we add the k' missing items from the under represented class so as to reach the desired balance. In particular, we pick those items that are the farthest away from the items of the under satisfied class picked from the diverse only algorithm. Finally, for each new element we add, we find its nearest neighbor in the over represented class and we remove it so as to make the size of the final set equal to k.

Efficiently computing the distance matrix that the algorithm needs was a a challenge, since there are $N = 23,708$ samples in $\mathbb{R}^{200\times200\times3}$. We use ℓ_2-distance as our distance function. In order to make the computation of this matrix feasible, we apply PCA to reduce the dimension of the data. In particular, we use the Incremental PCA and standarization techniques with a batch size equal to 1000 images; so the dataset was divided into 23 full-sized batches and one of 708 images. Then, we find the number of principal components (or eigenfaces) that sufficiently describe the data. Towards that direction, we seacrh for the top n eigenfaces that accumulate 96% of the total variance of the data. We find that $n = 330$ for the UTKFace dataset. Figure 1 shows the results of the PCA process:



(a) The mean face (b) The first 20 eigenfaces

Figure 1: Standarization and PCA results

We then compute the distance matrix and after running the first phase of the algorithm ignoring fairness and focusing on diversity only, we create a sample of 7813 male samples($\sim$ 52%) and 7187 female samples($\sim$ 48%). Because the ratio of the two classes is close to 1, we keep this sample as our *fair* and *diverse* dataset and we do not proceed with the second phase of the algorithm.

Finally, for the unbalanced and non-diverse dataset we create a $k = 15,000$ samples with 20% female and 80% male samples. Moreover, the $3,000$ female samples are chosen in a way so that their diversity score, thus the minimum pairwise distance, is low. We accomplish that by randomly picking the first element and then by adding the one that is the closest in the elements already picked. we repeat the procedure until k is reached. The male samples are collected by using a random sampling technique.

3.1.2 CNN architecture

Inspired by the architecture used by [6] for a similar task, we design a custom CNN network and train it from scratch. We use 3 convolutional layers and two fully connected layers as follows:

- Convolutional Layer 1: It consists of 16 filters of size 11×11 applied with stride $S = 3$ and padding $P = 0$. The we pass the output of size $16 \times 64 \times 64$ through a ReLU followed by a max pooling layer of size 2×2 with $S = 2$.

- Convolutional Layer 2: It consists of 32 filters of size 5×5 applied with stride $S = 1$ and padding $P = 0$ to the output of the previous layer of size $16 \times 32 \times 32$. The we pass the output of size $32 \times 28 \times 28$ through a ReLU unit followed by a max pooling layer of size 2×2 with $S = 2$.

- Convolutional Layer 3: It consists of 64 filters of size 3×3 applied with stride $S = 1$ and padding $P = 1$ to the output of the previous layer of size $32 \times 14 \times 14$. The we pass the output of size $64 \times 14 \times 14$ through a ReLU unit followed by a max pooling layer of size 2×2 with $S = 2$.

- Fully Connected Layer 1: It consists of 256 neurons followed by a ReLU unit and a dropout with $p = 0.5$.

- Output Layer: The output of the previous layer maps to $C = 2$ neurons and then is fed into a softmax classifier unit so as to produce the final prediction.

We initially planned to use more filters per convolutional layer as in [6] but after noticing that with this architecture we were able to fit the training data reasonably well, we chose to keep it simple.

For training and evaluation purposes, we split the training dataset of size $15,000$ into a training set of $12,000$ samples, a validation and a test set of $1,500$ samples each. Note that each split preserves the ratio of the two classes observed in the original dataset. This is important, especially in the case of unbalanced data, so that the validation and test sets contain samples of the minority class and we can check how well the model fits this class.

3.2. Model-level Approach

For this part of the project, we use a custom designed CNN architecture best suited for the UTKFace dataset. This creates the baseline for accuracy and it will be referred to as the **Baseline Model** throughout the paper. Initial testing suggests that most errors in gender classification are made for the under-represented groups. About 30% of the youngest group and 15% of the oldest group is misclassified. This leads us to the assumption that the dataset is most likely not fair w.r.t. the different age groups and we need more instances of the under-represented ones. With this in mind, we examine an aging software available online and create artificial data. Next, an ensemble of models are trained on different and fair subsets with respect to age and observe the effect this approach has in the accuracy of the

classifier. More specifically, we experiment with a collection of models trained on different fair and diverse subsets of the dataset to answer if they give more predictive power combined.
We will first go over the data sampling methods used to make the training sets for the ensemble of models and then briefly discuss the CNN architecture of the models.

3.2.1 Sampling Techniques

We use 3 methods of sampling. In each case, we select 3 disjoint samples of 7000 images to train 3 separate models that combinedly make the ensemble. The rest of the images are kept for testing.

- Random sampling: Random sampling without replacement is used.
- Stratified sampling: Keeping in mind the previous observation about under-represented age groups, the whole dataset is divided into 7 distinct categories or strata. These categories are based on age i.e. the data is sorted into 7 different age groups/categories. Data is picked at random from each of these categories to make samples of 7000 images.
- Stratified sampling with artificial images: Since the dataset under-represents people over 60, we inject 1500 artificial images into the main dataset. Change My Face[7] is an aging software, which was used to generate the aged versions of 1500 faces from the UTKFace dataset itself. The original dataset combined with artificial images are then sorted into different strata based on age and sampled like previously described.

3.2.2 CNN architecture

The network architecture we use in the model level approach is explained here. The baseline model as well the ensembles follow this architecture. The network comprises of only two convolutional layers and two fully-connected layers. Our choice of a smaller network design is to reduce the risk of overfitting and to have a reasonable training time. All three color channels are processed directly by the network. Images are first re-scaled to 144x 144 and fed to the network. The subsequent layers are defined as follows:

- 64 filters of size 2x2 are applied to the input in the first convolutional layer, followed by a ReLU activation, a max pooling layer taking the max of 2x2 regions, followed by a dropout layer with p=30% .
- 32 filters of size 2x2 are applied to the input in the second layer, followed by a ReLU activation, a max pooling layer of 2x2 regions, followed by a dropout layer with p=30%.
- The first fully connected layer contains 256 neurons, followed by a ReLU activation and a dropout layer.
- The last fully connected layer maps the previous output to the final classes for gender and these which is then fed to a sigmoid layer to get the final prediction.

4. Experimental Results

4.1. Data-level Approach

We conduct a series of experiments so as to evaluate the effect a *fair* and *diverse* dataset has on the performance of a gender classifier that uses the CNN architecture presented in section 3.1.2.

Towards this end, we first train a classifier on the *fair* and *diverse* dataset and then on the second dataset we generated. We evaluate the two classifiers and report their overall and per-class accuracy both during training and on the test set. We also report the precision and recall scores on the test set.

For completeness, we give the definition of the precision and recall metrics. In a binary classification setting, we consider that we have a positive and a negative class. Suppose that a classifier **correctly** labels a subset TN(True Negatives) of the test data as negatives and another subset of the test data TN(True Positives) as positives. There is also another set of data FN(False Negatives) that gets **falsely** predicted as negatives when they are positives. Finally, there is another subset FP(False Positives) that gets **falsely** predicted as positives when they are negatives. Then precision and recall are then defined as:

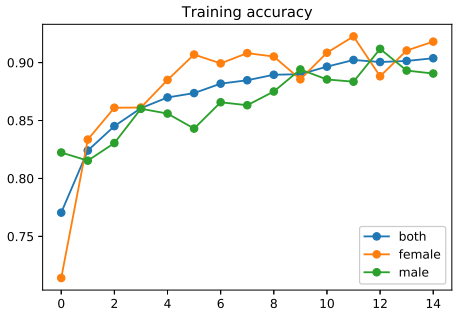
$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

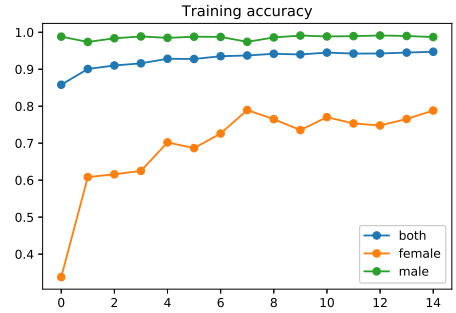
The range for the two metrics is from 0-1 while a good classifier has high precision and recall. We train the CNN model from scratch for 15 epochs on $12,000$ diverse samples using a batch size of 256 samples. We use Adam as an optimizer while we set the learning rate equal to $\text{lr} = 5e-4$ and the regularization strength equal to $\lambda = 2e-2$. We use the same hyperparameters and optimizer for both of the datasets so as to create the two classifiers. Let classifier A be the one that was trained on the *fair* and *diverse* set and classifier B the other one. The behavior of the two classifiers during the training phase is reported in figures 2, 3.

The results show that the classifier B has a lower accuracy in the female samples while there is also a large gap in the accuracy between the two classes. On the other hand,

classifier A is being relatively fair to how well it performs in both classes while demonstrating a better accuracy in female samples. From figure 3, we verify that the hyperparameters chosen demonstrate a good model behavior.



(a) Classifier A



(b) Classifier B

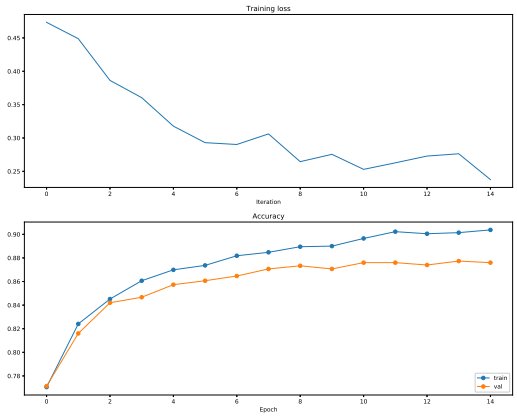
Figure 2: Per-class and overall accuracy for the two classifiers

Subsequently, we give the results for the two classifiers in terms of overall and per-class accuracy along with the precision and recall scores on the test set.

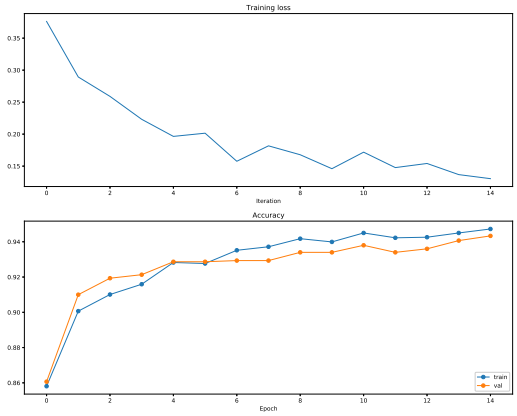
Metrics	Classifier A	Classifier B
Overall Accuracy	0.88	0.93
Female Accuracy	0.89	0.72
Male Accuracy	0.86	0.98
Precision	0.9	0.91
Recall	0.96	0.62

Table 1: Comparison between classifier A and B

We observe that classifier A has a good overall accuracy while both its precision and recall are high as desired. On the other hand, we see that for classifier B accuracy is not sufficient to define the quality of the model. Note that although its accuracy is high, its recall score is low which



(a) Classifier A



(b) Classifier B

Figure 3: Training loss and accuracy for the two classifiers

means it fails to correctly classify a significant proportion of the female samples. The fact that its accuracy is high was expected, since even by labeling all the test data as male, it would get an 80% accuracy since this proportion of the set was male.

We now check whether the naive over-sampling and SMOTE technique to helps in building a better classifier. Note that the over-sampling techniques are applied after splitting the data into train-test-val sets. Furthermore, in both cases we create a training set of size $12,000$ with $\sim 48\%$ female and 52% male so that we have a fair comparison with classifier A. In particular, we start with $9,600$ male samples and $2,400$ female samples. We perform random under sampling in the majority class so as to get $6,250$

samples. Then we proceed with the over sampling phase where we reach the $5,750$ female samples. The results in the training set are given in table 2.

Metrics	Over-Sampling	SMOTE
Overall Accuracy	0.94	0.93
Female Accuracy	0.85	0.87
Male Accuracy	0.96	0.95
Precision	0.82	0.8
Recall	0.74	0.85

Table 2: Results after over-sampling

Notice that although in both cases the recall score got better, the precision of the classifier dropped. This is a sign that the model starts to think that more samples are female, so it manages to classify correctly more female samples. At the same time though it starts misclassifying more male samples which shows that is overfitting towards the minority class.

We conclude that classifier A still performs better even after applying the over-sampling techniques since even the individual accuracy scores are comparable, although not the most sufficient metric. Moreover, classifier A has both high precision and recall and the results suggest that our initial hypothesis is true in this particular setting. We also show that dealing with imbalances is a hard problem. Thus, it might be worth focusing on designing novel sampling techniques that satisfy the characteristics a dataset should have so as to build robust classifiers.

4.2. Model-level Approach

For this part, experiments and their analyses are mostly straightforward. Ensembles made with different sampling techniques are compared with the baseline model. Our comparison metric here is accuracy i.e. the fraction of predictions that were correct.

The models are trained from scratch. 5-fold cross validation is used to evaluate the models. The baseline model is trained on 80% of the images and tested on 20%. Each ensemble is built by using 3 models. The final prediction of the ensembles are done by majority voting, which is one of the easiest ensemble techniques.

Figure 4 shows the difference in overall accuracies between the baseline model and the different ensembles. The baseline model seems to be performing better than all the ensembles. Next, we took a closer look into the age groups and their accuracies i.e. how each model was performing for people of different ages. Ensembles with random sampling has the lowest performance, so we will exclude that from further comparisons.

Figure 5 gives a bit more insight into accuracies across age groups. Although quantitatively the baseline model

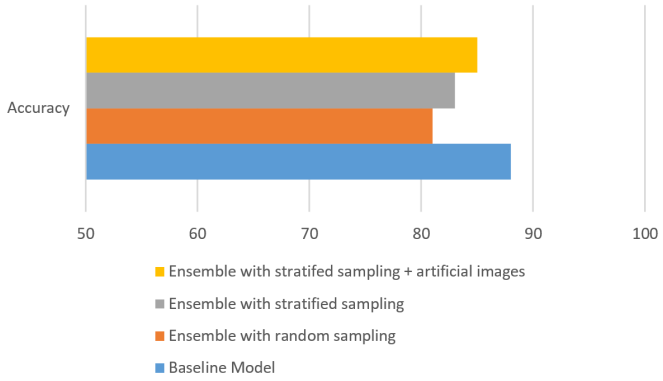


Figure 4: Comparison between models

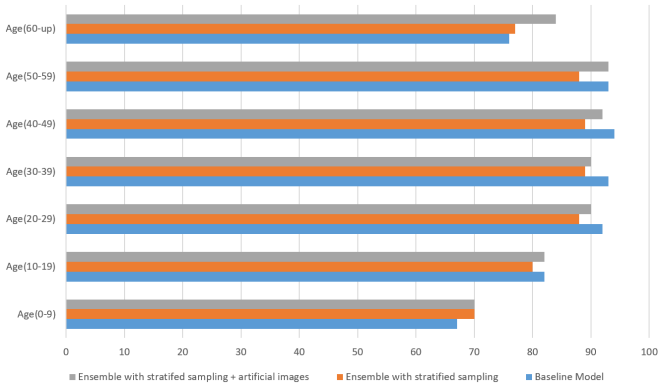


Figure 5: Accuracy across different age groups

has the best accuracy, it is not performing the same for all groups. It is clearly under-performing for people aged over 60. The reason for high overall accuracy relates to the fact that older people are under-represented and accuracy peaks for middle aged people who make up the majority of the original dataset. If we judge only by overall accuracy, this subtle detail goes unnoticed. Stratified sampling shows improvement for the under-represented group in question, but it still suggests that the group lacks in sufficient number of images. On the other hand, Stratified sampling with artificial images helps to make the accuracies across different age groups a bit more even, although the overall accuracy decreases slightly compared to the baseline.

Another thing to notice here is that, the accuracy for faces aged under 10 years is still low compared to the other groups. One plausible reason for this can be the lack of sufficient images, something we mentioned earlier. Unlike people over 60, we did not inject artificial images for this group. This can further suggest we need more data for this particular age group, whether real or artificial.

5. Conclusions

In this project, we assessed the significance of a fair and diverse training set and how it can improve a model compared to using an unbalanced or invariant set. To ensure fairness and diversity in the training set, we used MaxMin diversity metric and enforced balance in terms of gender. Since solely focusing on accuracy can often lead to inaccurate conclusions, we used precision and recall as additional metrics to compare performance. Based on these metrics, our results suggest that that the performance of the classifier trained on such a fair and diverse set outperforms a classifier trained on an unbalanced set even after using over-sampling techniques.

We also explored the idea of using model ensembles to boost accuracy and the use of artificially generated images when real data is scarce. We found that the inferior performance for under-represented groups cannot be improved with a model level approach such as an ensemble technique only. Unconventional methods like artificially generated images from software can come to aid when real labelled data is unavailable.

References

[1] G. Antipov, S.-A. Berrani, and J.-L. Dugelay. Minimalistic cnn-based ensemble model for gender prediction from face images. *Pattern Recognition Letters*, 70:59 – 65, 2016.

[2] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In S. A. Friedler and C. Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91, New York, NY, USA, 23–24 Feb 2018. PMLR.

[3] E. Celis, V. Keswani, D. Straszak, A. Deshpande, T. Kathuria, and N. Vishnoi. Fair and diverse DPP-based data summarization. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 716–725, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[4] M. Duan, K. Li, C. Yang, and K. Li. A hybrid deep learning cnn–elm for age and gender classification. *Neurocomputing*, 275:448 – 461, 2018.

[5] K. Kärkkäinen and J. Joo. Fairface: Face attribute dataset for balanced race, gender, and age, 2019.

[6] G. Levi and T. Hassner. Age and gender classification using convolutional neural networks. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) workshops*, June 2015.

[7] C. M. F. Ltd. https://changemyface.com/, 2017.

[8] D. M, J. HV, P. E, and S. J. Diversity in big data: a review. *Big Data*, 5:2:73–84, 2017.

[9] M. Merler, N. Ratha, R. S. Feris, and J. R. Smith. Diversity in faces, 2019.

[10] S. Y. Zhang, Zhifei and H. Qi. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.