Advancing Ethical AI : Fairness, Diversity, and Privacy in Generative Modeling

Mariia Zameshina

Université Gustave Eiffel

Advancing ethical AI: fairness, diversity, and privacy in generative modeling

Thèse de doctorat de l'Université Gustave Eiffel

École doctorale n°532, Mathématiques et Sciences et Technologie de l'Information et de la Communication, MSTIC

Spécialité de doctorat: Informatique

Unité de recherche : Labortoire d'Informatique Gaspard-Monge, LIGM

Thèse présentée et soutenue à l'Université Gustave Eiffel, le 26 mars 2024, par:

Mariia Zameshina

Composition du Jury

Julia Kempe Professeur, New York University	Examinateur
Liva Ralaivola Professeur, Aix-Marseille Université VP Research, Criteo AI Lab	Rapporteur
Jean-Michel Loubes Professeur, Université Toulouse Paul Sabatier	Rapporteur
Emily Wenger Chercheuse, Facebook AI Research (Meta)	Examinateur
Stéphane Lathuilière Maître de conférence, Telecom Paris	Examinateur

Encadrement de la thèse

Laurent NAJMAN Professeur, Université Gustave Eiffel	Directeur de thèse
Olivier TEYTAUD Chercheur, Facebook AI Reserach (Meta)	Co-Directeur de thèse

Contents

Abstract

This thesis explores the ethical considerations of generative modeling. Specifically, we investigate the issues of privacy, diversity, and fairness in the context of image generation. We propose and evaluate several methods for enhancing image diversity, including Diverse Diffusion, which encourages the generator to produce diverse images that span beyond gender and ethnicity, including color diversity. We also introduce PrivacyGAN, a robust generative image privacy method that protects sensitive information in generated facial images. Additionally, we design unsupervised fairness algorithms that address fairness issues and mode collapse in generative modeling.

Résumé

Cette thèse explore les considérations éthiques de la modélisation générative. Plus spécifiquement, nous étudions les problèmes de confidentialité, de diversité et d'équité dans le contexte de la génération d'images. Nous proposons et évaluons plusieurs méthodes pour améliorer la diversité des images, y compris la Diffusion Diversifiée, qui encourage le générateur à produire des images diverses qui vont au-delà du genre et de l'ethnicité, incluant la diversité des couleurs. Nous introduisons également Privacy-GAN, une méthode robuste de protection de la vie privée pour les images génératives, qui protège les informations sensibles dans les images faciales générées. De plus, nous concevons des algorithmes d'équité non supervisés qui abordent les problèmes d'équité et de collapsus de mode dans la modélisation générative.

Acknowledgements

I would like to express my sincere gratitude to ANRT and Meta (Facebook AI Research) for their financial support of my thesis, which has been essential in enabling my research endeavors.

Special appreciation is extended to my supervisors, Laurent Najman and Olivier Teytaud, for their invaluable guidance and unwavering support throughout my research journey. Their expertise and insightful advice have been fundamental in advancing my academic progress.

My gratitude also goes to my fellow PhD collaborators, Marlene and Guillaume, for their significant contributions. I am equally thankful to the other PhD students at Meta, University Gustave Eiffel, and Paris Telecom for the enriching and insightful discussions that have greatly enhanced my PhD experience. Thank you Marta, Yamna, Caroline, Sarah, Baptiste...

I would like to acknowledge the additional guidance provided by other researchers from Meta, namely Sam, Vasil, Emily and Artyom. Their direction and insights in furthering my reasearch and career have been immensely valuable and deeply appreciated.

Additionally, I would like to give special recognition to my mentors, Rui and Eugene, for their mentorship and support. Their perspectives have been crucial in the development and shaping of my research work.

Furthermore, I extend my thanks to my French teacher, Marie-Laure, for her assistance in improving my French and helping me adapt to day-to-day life in France.

Lastly, I express my heartfelt gratitude to my family and friends: my parents Elena and Aleksandr, grandparents Anatoly and Stalina, my partner Sibasish, my brother Andrey, my friend Elina, and the Grenoble squad - Aleksandr, Yevheniya, and Hanna - for their constant encouragement and support. Their unwavering belief in my abilities has been a continual source of strength and motivation throughout this journey.

1

Introduction and Literature Review

1.1 Research Questions and Objectives

In an era where artificial intelligence is reshaping our interaction with digital media, this thesis is dedicated to unraveling the complex ethical dimensions intertwined with generative image synthesis.

Central to our investigation is the exploration of using generative models as tools for enhancing privacy, alongside examining possibilities in promoting diversity and fairness. Through research and comprehensive analysis, we aim not only to deepen our understanding of these applications and ethical concerns but also to contribute significantly to the development of responsible generative modeling practices.

The primary objectives of this research are to:

- Investigate and articulate the potential of generative image synthesis in enhancing privacy, while also considering the concerns of diversity, and fairness.
- Design and evaluate novel algorithms and methodologies that utilize generative modeling as a means to safeguard privacy and mitigate ethical concerns within the domain of generative modeling.

1.2 Plan of the chapter

This chapter is structured as follows:

- **Generative Modeling and its Applications (Section 1.3)**:

An introduction to the fundamental concepts and applications of generative modeling is provided, emphasizing the definitions that we further use in this thesis.

- **Ethical Concerns of Generative Modeling (Section 1.4)**:

  Exploration of the ethical challenges, including bias, diversity, in generative modeling and of the possibility to use generative modeling for protecting privacy.

- **Fairness in Generative Modeling (Section 1.5)**:

  Analysis on addressing fairness issues in generative models, including examining Mode Collapse and exploring solutions.

- **Diversity in Generative Modeling (Section 1.6)**:

  Discussing the critical role of diversity in generative models, addressing class imbalances, color variations, and conceptual diversity.

- **Generative Modeling for Privacy Preservation (Section 1.7)**:

  Elaborating on the use of generative modeling techniques to enhance online privacy , and evaluating various methods and their implications.

- **Thesis Structure (Section 1.9)**:

  Outlining the complete structure of the thesis, this section details the focus and contributions of each chapter.

1.3 Generative Modeling and its Applications

This section delves into the expansive landscape of generative modeling, elucidating its core principles and exploring a spectrum of applications. Within this realm, we discuss prominent techniques, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Latent Diffusion Models (LDMs).

In recent years, the field of generative modeling, particularly in image synthesis, has undergone transformative advancements.

However, this newfound power comes with the problems such as hallucinations and poor commonsense reasoning [OHP+23] and ethical challenges related to fairness, diversity, and privacy. The papers exploring fairness, diversity, and privacy in generative modeling are critical in navigating this intricate terrain. They showcase the potential biases embedded in models, underscore the significance of diverse representation, and highlight the importance of preserving individual privacy.

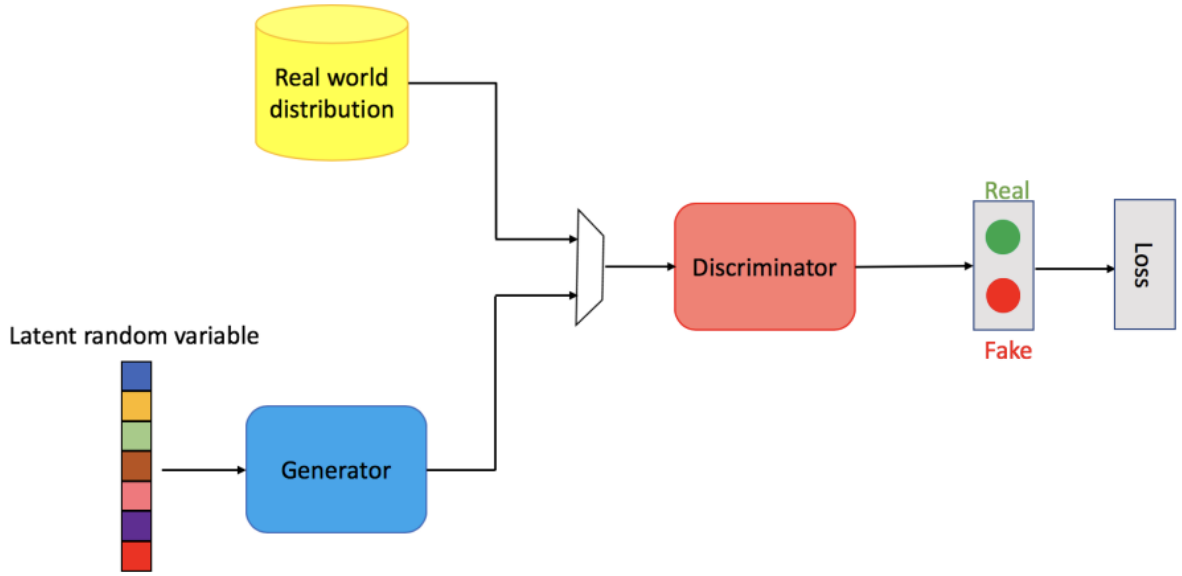

Figure 1.1: Generative Adversarial Networks structure. It comprises of generator that aims to generate images resembling dataset instances and discrimnator that identifies whether an image is real or fake. Image credit: [CG23]

Generative Adversarial Networks (GANs), introduced by Goodfellow et al. in 2014 [GPAM+14], operate through an adversarial training mechanism. In this setup, a generator crafts data instances, and a discriminator distinguishes between genuine and generated samples. This interplay refines the generator's ability to create data, resulting in the synthesis of realistic data instances. The GAN structure is presented in Figure 1.1.

The impact of GANs extends to image synthesis, exemplified by StyleGAN [KLA19a], renowned for generating realistic high-resolution images with diverse styles. Beyond traditional domains, GANs have found applications in the creative realm, contributing to the generation of art [ELEM17] and music [YCY17].

Variational Autoencoders (VAEs) represent another facet of generative modeling. Comprising an encoder and a decoder, autoencoders offer the ability to reconstruct images. VAEs extend this framework with probabilistic components, enabling the generation of diverse and meaningful samples from a learned latent space. In the case of VAEs latent space is represented by the mixture of the distributions. The VAE structure is presented in Figure 1.2.

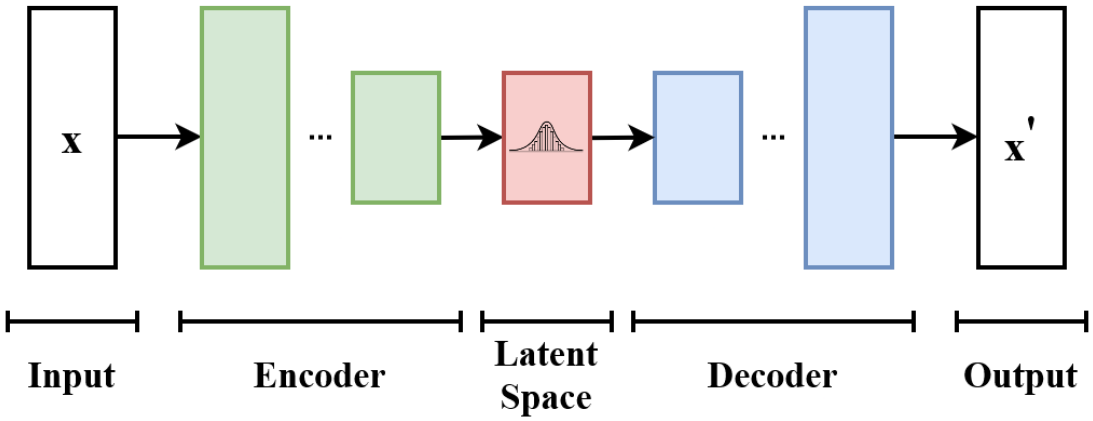

Figure 1.2: Variational Autoencoder structure. The model recieves data instance x, compresses it to the latent space and from there decodes x' that is similar to the original data instance x. Image credit: https://en.wikipedia.org/wiki/Variational_autoencoder

Latent Diffusion Models (LDMs) mark an advancement in artificial intelligence, particularly in the generation of detailed images from textual descriptions. Notable examples include DALL-E [RPG+21] and Stable Diffusion [RBL+22]. In diffusion models, the forward diffusion process consists of multiple steps, each of which is adding Gaussian noise to the data sample. The reverse diffusion process aims to recreate the true sample from the noise. In the case of LDMs, the diffusion process is done not in the pixel space of an image but in the latent space. The LDM structure is presented in Figure 1.3.

Further, we use the following definitions.

Fairness in Generative Modeling: In the realm of generative modeling, fairness pertains to the unbiased generation of data, ensuring that the model does not perpetuate or amplify existing biases present in the training data.

Mode collapse is an effect in generative modeling, where the model generates data lacking the representation of different groups or characteristics encoded in the training dataset.

Diversity in Generative Modeling: Diversity in generative modeling refers to the ability of a model to produce a wide range of outputs, avoiding over-representation of certain classes or features and promoting inclusivity in the generated data.

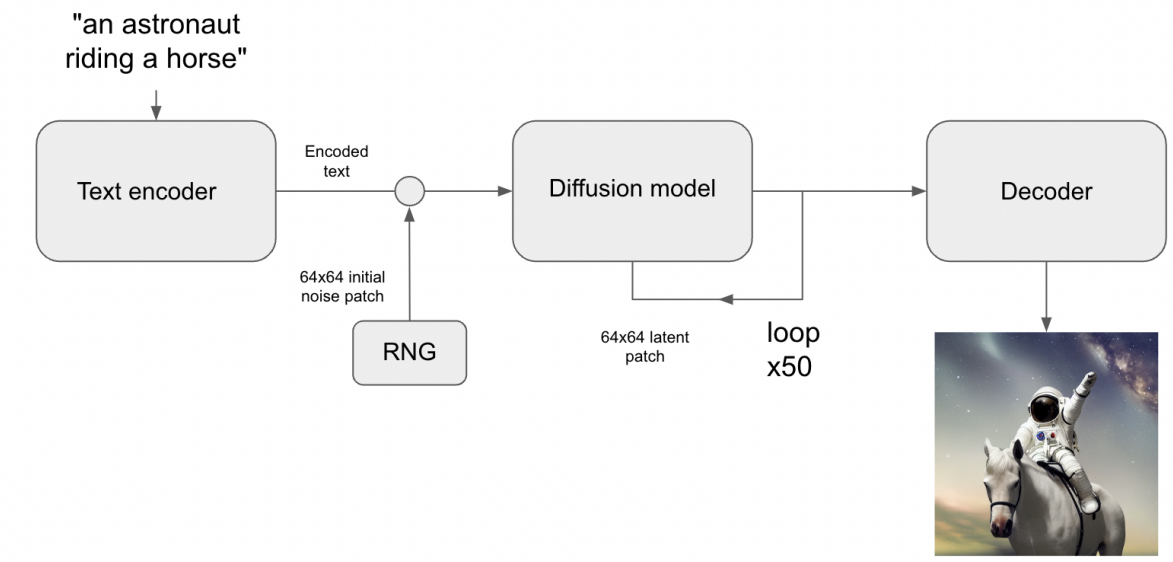

Figure 1.3: Latent diffusion model structure. It consists of an encoder, that encodes the text into the latent space, a diffusion model that is run in this latent space and a decoder, that recreates the image. Image credit: https://keras.io/examples/generative/random_walks_with_stable_diffusion/

Image anonymization refers to the class of techniques that fully remove any identity information from the facial images [WSZZ23].

Privacy preservation: In contrast to anonymization, by privacy preservation in this thesis we mean image privacy against unknown recognition systems [WSZZ23, SNN23].

1.4 Ethical concerns of generative modeling

Generative models, like the ones we discussed above, bring up important ethical questions. One primary concern is bias within these models. Section 1.5 explores the research related to bias, highlighting challenges and suggesting ways to mitigate these problems.

Beyond bias, ethical issues with generative models extend to diversity. Section 1.6 delves into potential issues where generative models might not adequately capture different colors or a variety of ideas in their outputs. This section aims to understand challenges in ensuring that generative models create outputs that are rich and inclusive.

Generative models bring up more ethical questions besides bias and diversity. Section 1.7 looks into an interesting aspect: using generative models as tools to solve ethical problems. This section explores how we

can use generative models for privacy protection online, giving us a two-sided view of their role in ethical discussions.

Collectively, these chapters contribute to the ethical foundations of responsible generative modeling, ensuring that AI technologies align with principles of fairness, diversity, and privacy.

Ongoing research is essential to stay abreast of new challenges and ideas in the field of image generation. The subsequent sections provide an overview of research focusing on different ethical aspects of generative modeling.

1.5 Fairness in generative modeling

The intersection of ethics and artificial intelligence places a spotlight on fairness in generative modeling. Issues such as Mode Collapse (MC) [RW18], where certain classes become exceedingly rare, underscore the importance of addressing fairness concerns. Chapter 2 explores fairness in an unsupervised fashion in the context of generative modeling.

1.5.1 Fairness

Fairness has become prevalent at the intersection of ethics and artificial intelligence. Various forms of fairness are critical in online media [HAM+19]. In our work, we consider fairness for generative modeling. More precisely, when modeling the probability distribution of faces, we typically observe that classes already rare in the dataset become even rarer in the model. This phenomenon is called Mode Collapse (MC) [RW18], and for sensitive variables, it is one of the fairness issues.

There are many facets to fairness. An algorithm may be considered to be fair if its results are independent of some variables, particularly for sensitive variables.

Fairness [PS20] can be measured in terms of separation, i.e., whether the probability of a given prediction, given the actual value, is the same for all values of a sensitive variable. The measurement can also be rephrased in terms of equivalent false negative and true negative rates for all classes.

A distinct point of view is sufficiency: sufficiency holds if the probability of actually belonging to a given group is the same for individuals from that group and with different sensitive variables.

Another point of view is independence, i.e., when the prediction is statistically independent of sensitive variables.

Because it is known that the many criteria for fairness are contradictory, it is necessary to design criteria depending on the application. In the

present chapter, we consider the case in which the goal is to preserve some frequencies.

Here, we consider the context of generative modeling. There is a model trained on data, and we want this model to satisfy some requirements on frequencies: for every class, we would like the frequency to match some target frequency. Typically, for simplicity in the present chapter, the target frequency is the frequency in the original dataset: however, the methods that we propose can be adapted to other settings.

1.5.2 Generative modeling: fairness and mode collapse

There are many measures of fairness, even in the specific case of generative modeling [TC21]. The main criterion is whether all classes are correctly represented. It is known that modeling frequently decreases the frequency of rare classes (i.e., mode collapse). In addition, improving the image quality (for each image independently) aggravates the diversity loss [SJJ19]. For a conditional generative model, there is sometimes a ground truth. For example, in super-resolution, we want the reconstructed image to match the sensitive variables of the ground truth as closely as possible. This case became particularly critical since, e.g., [Tru20]: a pixelized version of Barak Obama can be "depixelized" to be that of a white man. [XYZW18] points out the importance of fairness in the design of Generative Adversarial Networks (GANs) before applying them, for example as an early stage before supervised training. For addressing fairness issues, a possibility is to increase editability: [HPK+20] disentangles latent variables for separating editable and sensitive parts. Some works focus on measuring fairness, for example, [KLRS17] uses causal methodologies for measuring fairness in a counterfactual manner. Fairness can be integrated directly into the training: [SHCV18] focuses on training a GAN while protecting some variables.

There are multiple works related to fairness in generative modeling. [CGS+20] increases fairness in GANs in a supervised manner, i.e., given the sensitive attributes. [SHCV19] targets and improves the fairness of generated datasets. More similar to our work, [JKH+21] focuses on uncertain sensitive variables, and [KAH+21] adds a bias in a GAN for mitigating fairness issues. In the same fashion as the present work, [TSZ20] considers biasing a GAN without any retraining. In chapter 2 we focus on generically (i.e., independently of the application, data, and model) correcting for potential bias present in a generative model, *without knowing the sensitive variables*. The critical point is that sensitive variables seem to often come up as a surprise: typically, people do not decide to create an unfair

algorithm actively. For example, in [Ple21], the designers of the faulty soap dispenser had just not imagined that it might fail on black skins. Also, there may be relevant sensitive variables that have not been initially considered: ethnicity or gender are obvious sensitive variables, but aesthetics, body mass index, social origin, or even the quality of the camera, geographical origin, also matter.

1.6 Diversity in generative modeling

While mitigating bias in generative modeling remains a priority, there is an equally crucial need to foster diversity within this realm. The diversity challenge extends beyond addressing class imbalances (such as gender or ethnicity imbalance) and has other aspects such as color variation (e.g., ensuring the model generates not only red roses but also roses of other colors) and conceptual diversity (considering different contexts like roses in vases, held in hands, or featured in art pieces). This section delves into methodologies aimed at enhancing diversity in latent diffusion models.

While facial images in the training datasets can often be imbalanced, they typically consist of images depicting people of different genders. In contrast, for the generated images it is not always the case. Here, we provide an example of a mode collapse in MidJourney (https://www.midjourney.com).

Latent diffusion models have gained significant attention for text-to-image generation, with DALL-E [RPG+21] and Stable Diffusion [RBL+22] being some of the most prominent examples. While they have shown impressive results in generating high-quality images from textual descriptions, there have been concerns regarding potential causes of their usage for various sensitive applications.

To prevent the potential misuse of these models, a few recent efforts have been made. For instance, [CBLC22] investigated the efficacy of Stable Diffusion for the medical imaging. Additionally, [SSG+22] explored image-retrieval frameworks to detect content replication in generated images. [SLYZ22] developed a classifier to trace back generated images to their source models, ensuring accountability for creators. Furthermore, [KRMA23] created faithful diffusion that selects images corresponding to a prompt.

Bias can arise from various sources, such as the training data or the algorithmic biases inherent in the model architecture. [SHK22] shows that text-to-image models pick up cultural biases linked to various Unicode

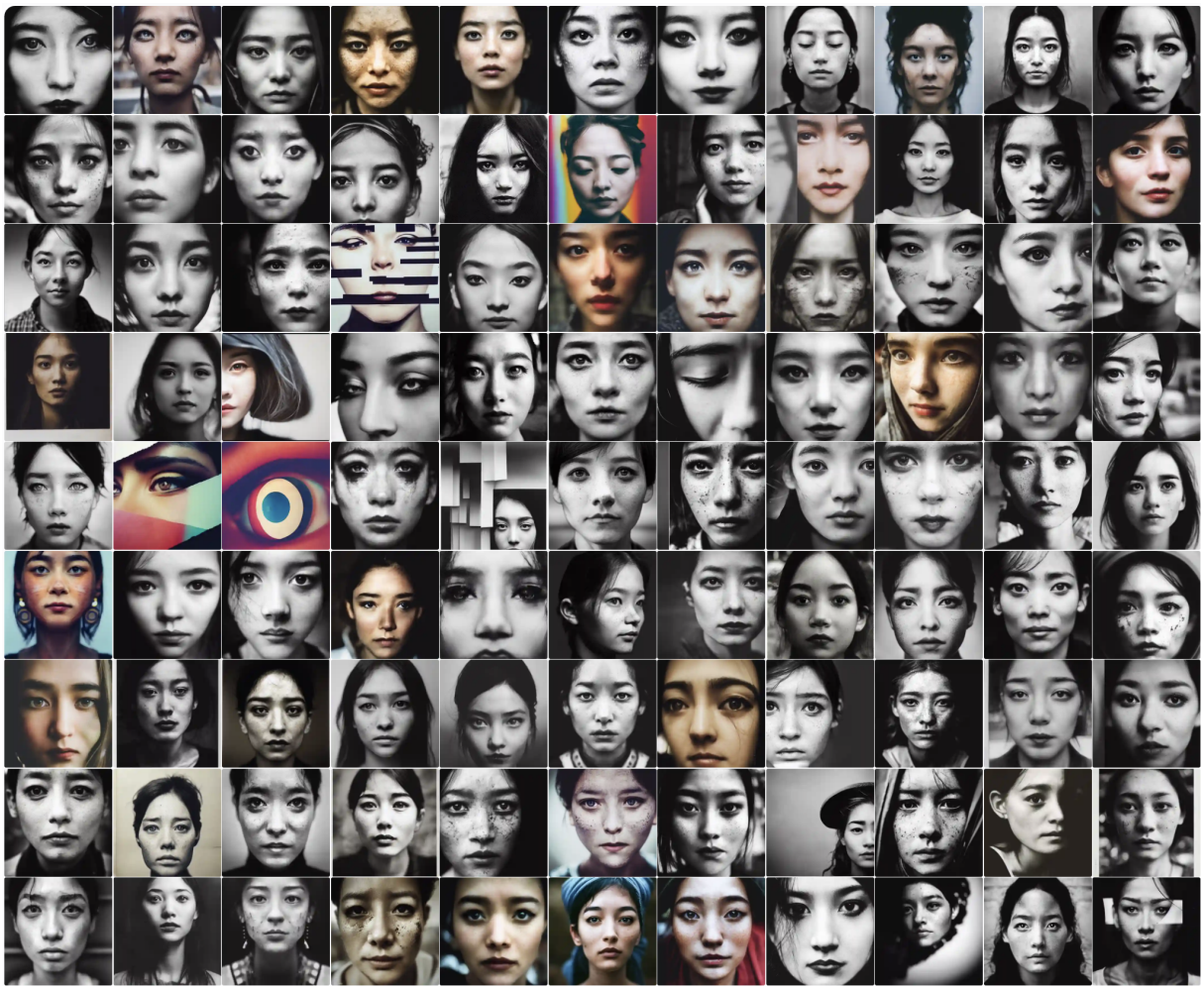

Figure 1.4: Images generated by Midjourney for the prompt 'Person entire face crop'. Here we can see that the vast majority of the images represents white or Asian females, while it was not originally specified in the prompt.

scripts. While increasing training data quality remains a significant concern and solutions such as [SFH+23] have been proposed to tackle dataset level bias, it is also important to ensure that the diffusion-based methods do not amplify any biases present in the training data.

The lack of diversity problem was addressed in [Ho23], [BKD+23] and [FKN23]. There, the authors notice that images generated by Stable Diffusion lack diverse cultural representation and are prone to gender stereotypes. [Ber22] highlights the need for algorithmic adjustments in generative models to increase the diversity of their output for multi-solution tasks. It also proposes a framework that integrates automated machine learning with computational creativity to automate key tasks in artistic pipelines and increase the creative autonomy of computational agents. Further, [The23] argues that Stable Diffusion v1-4 violates demographic parity in generating images of a doctor given a gender- and skin-tone-neutral prompt. The author observed that the model is biased towards generating images of perceived male figures with lighter skin, with a significant bias against figures with darker skin, as well as a notable bias against perceived female figures. According to [Bar22], AI image generators often display gender and cultural biases. Stable Diffusion as other models has inherent biases from the training datasets. It was found, for instance, that Stable Diffusion depicts all engineers as male despite women making up around 20% of people in engineering professions. According to [CLG+23] Stable Diffusion bears the nontrivial biases due to learned relations between not necessarily related concepts like "millennials" and "drinking".

There are recent methods focusing on diversity for Stable Diffusion generated images as well. Most of them focus on specific domains to broaden the image variability. For instance, [SWT+23] and [BYMC22] report that adding supporting context to text-to-image models prompts increases diversity in both general and human-specific fashion. In [SBAD+23] the authors report increase in generation of rare-concept images following the seed manipulation. [KKSY23] trains prompt embeddings that would guide to generate images fairly according to the set of sensitive attributes such as gender and ethnicity.

Promoting novelty and diversity is a challenge that exists both for image and text generation. In [XRLS18] the authors emphasize the importance of producing novel and diverse textual outputs. By leveraging a language-model-based discriminator, their model DP-GAN assigns high rewards to text that exhibits both novelty and fluency. Our approach for image gen-

eration also goes beyond addressing sensitive fairness concerns and opens new avenues for creative expression, as we focus not only on representation of people with different ethnicities and genders, but also, as our approach is unsupervised, on diversity of color and other dissimilarities of images in a batch.

It is common to generate multiple image versions using latent diffusion models. Even in non-sensitive scenarios, having diverse outputs is essential. A collection of similar images holds little advantage over a single image. Therefore, we propose a method that encompasses various domains, including faces, cars, animals and more to enhance diversity in text-to-image generation. By increasing diversity within batches, we can reduce the number of required generation iterations. For instance, by increasing the probability of obtaining satisfactory images from 1% with a vanilla Stable Diffusion to 5% using our method, the expected number of generated batches before satisfaction decreases by a factor 5.

1.7 Generative Modeling for Privacy Preservation

While ethical concerns related to generative modeling have been discussed in the preceding sections, there exists an intriguing prospect to leverage generative modeling for ethical purposes. This section, in particular, delves into the application of generative modeling for preserving user privacy online.

Individuals often share personal photos on various social media platforms, which facilitates communication and connection with family, friends, colleagues, and customers. Unfortunately, a significant drawback of this practice is that it can sometimes be possible to identify individuals social media accounts by taking their picture in public [Sam19] or by comparing their dating app photos to their business-related social media profiles. This is often made possible by the existence of datasets collected by scraping social media platforms. While face detection systems can be used by the government for criminal identification purposes, they also present opportunities for both internal and external misuse [WLVGP09], including enabling stalkers to track their victims [Har22]. Consequently, sharing real facial images publicly over the internet may compromise users privacy.

Multiple initiatives are dedicated to enhancing image and video privacy on the internet. One of the prominent groups is centered around **anonymization methods**, which involve altering users pictures to resemble those of other individuals. For instance, [KY19] proposed a privacy-

preserving adversarial protector network (PPAPNet) as an image anonymization tool. PPAPNet transforms an image into another synthetic yet realistic image while remaining immune to model inversion attacks [WFL+21]. Anonymization techniques may also preserve key characteristics such as background, emotions, and facial feature movements [HMB+23, HML19, BTPS23, HL23]. These techniques are valuable when the objective is to maintain realistic appearances without the need to recognize individuals in photos or videos. Such approaches are particularly useful, for example, for maintaining anonymity while expressing opinions on video-sharing platforms. For instance, [GWT19] achieves the objective of decorrelating the identity while retaining the perception (pose, illumination, expression). Some of these methods change only parts of the face. As an example, in their work [QNS+22], the authors suggest utilizing generative techniques to enhance images that have been intentionally blurred or have had the subject's eyes obscured beforehand.

In the overview [WSZZ23], the authors tackle a challenge in the design of Anti-Facial Recognition (AFR) systems: finding a balance between **privacy, utility, and usability**. They categorize AFR systems based on their target components, ranging from data collection and model training to run-time inference, all with the shared objective of thwarting successful recognition by unauthorized or unwanted models. Moreover, the authors stress the user preference for privacy tools with minimal overhead, a concept underscored by studies such as [SBBR16] and [DSDN19]. These findings highlight the significance of delivering protection against image recognition systems while mitigating any adverse effects on the user experience, a goal that many present anonymization methods struggle to attain. While certain attributes of images, such as gender, ethnicity, and facial expressions, can be retained through specific anonymization techniques, the resulting modified images frequently lack practicality for users. As a result, even though these images maintain crucial visual characteristics, the individual's identity within them may undergo substantial alterations, ultimately rendering them unidentifiable to acquaintances and family members.

In an effort to achieve a balance between utility and privacy protection, another area of research focuses on **obscuring facial images to maintain human recognizability while creating difficulties for neural networks to decipher**. Generally, these methods involve introducing precisely crafted pixel noise, causing the neural network to misclassify the im-

age. These pixel-level perturbations have effectively challenged diverse image recognition neural networks. The dilemma of balancing privacy maintenance with recognition assurance of data-poisoning methods like Fawkes [SWZ+20a] and Lowkey [CGF+21] is discussed in detail in [RDT21].

As an alternative approach to safeguarding images against unauthorized identification while preserving their utility for the users, one can consider **adversarial examples**. [QNS+22] demonstrated that makeup transfer can be an effective means of countering various face recognition systems. However, this method has limitations, as the model's performance may be inconsistent between male and female images due to an imbalance in the makeup transfer training dataset. This method is also ineffective in cases where the face cannot be found on an image, as it is not possible to transfer makeup in this case. Additionally, some individuals may find the use of makeup transfer images unacceptable.

1.8 Ethics Statement

Our research is conducted with a commitment to ethical principles.

The facial images that we use either contain individuals who consented to participate in the study (i.e., authors of the papers), are taken from publicly available datasets, or consist of AI-generated art that does not represent real humans.

The code for generating images and the evaluation methods are either made openly available or described in detail, enhancing transparency and facilitating reproducibility.

This thesis contributes not only to the technological aspects of generative modeling but also to the ethical discourse surrounding AI. By addressing topics such as fairness, diversity, and privacy, we aim to foster the responsible development and application of AI technologies.

1.9 Thesis structure and contributions

1.9.1 Chapter 1: this chapter

Chapter 1 provides introduction to the current thesis and a review of the existing literature surrounding generative modeling, with a particular emphasis on the ethical concerns raised in recent research.

1.9.2 Chapter 2: fairness in GANs

Chapter 2 identifies fairness challenges at the intersection of ethics and artificial intelligence, specifically focusing on generative modeling.

The work concentrates on the phenomenon of Mode Collapse, where rare classes in the dataset become even rarer in the generative model. This is particularly observed in the context of image generation, specifically for faces.

To sum up our contributions,

- We provide a novel solution for generic correction of biases in the context of generative modeling, independent of sensitive variables (unsupervised).
- We also evaluate how techniques for improving image quality might degrade fairness, and introduces methods to mitigate such issues.

These contributions collectively form a foundation for addressing fairness challenges in generative modeling, providing both theoretical insights and practical tools for improving diversity and mitigating biases. We published our findings in [ZTT+22].

1.9.3 Chapter 3: diversity in latent diffusion models

Chapter 3 presents a Diverse Diffusion, a modification to the Stable Diffusion algorithm for diverse image generation.

To sum up our contributions,

- We introduce a novel and general unsupervised technique based on re-weighting applicable to existing text-to-image models to increase image diversity;
- We conducted experiments demonstrating the diversity advantages of the proposed approach applied to stable diffusion.
  - We demonstrated notable improvements in the representation of underrepresented categories across different ethnicity/gender pairs and colors, promoting diversity and inclusion in image generation.
  - We also observed the improvement for general metrics such as LPIPS and demonstrated that our method is more effective than state-of-the-art prompt manipulation techniques.

We published our findings in [ZTN23].

1.9.4 Chapter 4: privacy using generative models

Chapter 4 presents a facial privacy preserving method, PrivacyGAN.

Summary for Chapter 4 contributions:

- Proposing an innovative approach for facial image privacy using generative methods and a distant target image idea introduced by Fawkes [SWZ+20a].
- Introducing a novel privacy evaluation method based on the proximity of dataset images in an embedding space to a modified "private" image.
- Creating a new facial image dataset extracted from Casual Conversations' dataset videos.
- Evaluating privacy against various embedding methods, including transfers to embeddings not used in our privacy method.
- Conducting human evaluations to assess image quality for both state-of-the-art and newly proposed privacy methods.

We published our findings in [ZCTN23]

1.9.5 Chapter 5: conclusions

Chapter 5 is focused on the conclusions from the obtained results, summarizing Chapters 2, 3 and 4, and provides future research plans.

2

Fairness in generative modeling

2.1 Introduction

Fairness has become prevalent at the intersection of ethics and artificial intelligence. In the present work, we consider fairness in the context of generative modeling.

More precisely, when modeling the probability distribution of faces, we typically observe that classes already rare in the dataset become even rarer in the model. This phenomenon is called Mode Collapse (MC) [RW18], and for sensitive variables, it is one of the fairness issues.

The contributions of this chapter include

- proposing tools based on statistical reweighting (Sections 2.3.1 and 2.3.2) and on user feedback (Section 2.3.3) for mitigating fairness issues (such as Mode Collapse) in generative modeling;
- evaluating how quality improvement techniques for image generation degrade fairness and how our proposed methods can mitigate such issues.

It is known that the many criteria for fairness are contradictory, that is why it is necessary to design criteria depending on the application. In the present chapter, we consider the case in which the goal is to preserve some frequencies.

Here, we consider the context of generative modeling. There is a model trained on data, and we want this model to satisfy some requirements on

frequencies: for every class, we would like the frequency to match some target frequency. Typically, for simplicity in the present work, the target frequency is the frequency in the original dataset: however, the methods that we propose can be adapted to other settings.

Our goal in this chapter is to have a generic correction of biases (in the context of generative modeling) independent of the sensitive variables.

The first proposed method (Sections 2.3.1 and 2.3.2):

- is not only for the fairness issues regarding sensitive variables: we also preserve diversity for more classical diversity issues such as Mode Collapse.
- does not need any retraining.
- is more or less effective depending on cases but is designed for (almost) never being detrimental (Section 2.4.2).

The second proposed method, which can be combined with the previous one,

- proposes several image generations and then
- lets the user choose one of the images among all the images generated during the previous stage.

Therefore, the user experience is modified: we expect the user to assist the method by actively selecting relevant outputs. Contrary to the generic method proposed above, which we will implement thanks to reweighting, the new approach is not a drop-in replacement. Moreover, this also does not need retraining.

2.1.1 Outline

Section 2.2 presents tools useful for the present work:

- Use of Image Quality Assessment (IQA) to improve image generation (Section 2.2.1): we connect this method to our research by investigating how much this quality improvement degrades fairness and how our proposed methods can mitigate such issues.
- Reweighting via simple rejection sampling to improve fairness and reduce Mode Collapse when the variables used for computing the reweighting values are correlated to the target sensitive variables (Section 2.3.1).

Section 2.3 presents our proposed algorithms:

- Reweighting as above, but with reweighed variables unrelated to target classes (Section 2.3.2). This second context is therefore applicable when we do not know the target classes. We propose a method which is a drop-in improvement of an arbitrary generative model: as soon as we have features and a generative model, we can apply Alg. 1.
- Multi-objective optimization, through computation of several solutions (typically Pareto fronts), to mitigate diversity loss by providing more frequently at least one output of the category desired/expected by the user.

Section 2.4 is a mathematical analysis. Section 2.5 presents experimental results.

2.2 Preliminaries

2.2.1 Correlations image quality / sensitive variables

We investigate the known correlation between the estimated quality of an image and its membership to a frequent class [SJJ19, MDH+20].

In order to demonstrate that this is easily observable, Table 2.1 presents the rank correlation between the aesthetic quality of an image and the logit of that image for each of four classes of individuals. We note that the most positively correlated class is the most frequent.

Our interpretation is that the technical quality of generated images is higher for the most frequent classes, influencing the aesthetics score.

2.2.2 Image generation: GAN, PGAN, and EvolGan

Our work specializes in image generation, and in particular on faces. We use the following image generation tools.

- Our baseline GAN is Pytorch GAN Zoo ([Riv19], based on progressive GANs (PGANs) [KALL18a]).
- We also use EvolGan [RTH+20], which improves Pytorch GAN Zoo by biasing the random choice of latent variables z using K512 [HLSS20].
- We use three configurations of EvolGan, as it uses as a budget the number of calls to the original GAN; the three configurations then correspond to budgets 10, 20, and 40 (named $EG10$, $EG20$, and $EG40$ respectively).

Class	A	B	C	D
Frequency	17.8%	**52.2%**	17.5%	12.4%
Rank-correlation AvA	-0.07	**0.22**	-0.11	0.06
Rank-correlation K512	-0.02	**0.16**	-0.08	0.02

Table 2.1: For four distinct classes of individuals A, B, C and D (obtained using R), we present the rank-correlation of the frequency of that class with AvA and K512 scores respectively.
AvA and K512 are visual quality estimators, dealing with aesthetics and technical quality respectively. Visual quality assessment is a task fairly independent of semantics and therefore should exhibit little if any ethnicity-related biases.
Dataset: faces generated by StyleGan2 (see `thispersondoesnotexist.com`).
Classes: ethnicity evaluated by R (see R in Table 2.2).
Observation: the biggest class has the strongest, positive correlation.

Besides the one based on a random search, EvolGan has an option for CMA search [HO03] and PortfolioDiscrete-$(1+1)$ (i.e. the variant of the Discrete $(1+1)$-ES as in [DL16]): we also employ these variants, with notation respectively EG-CMA-10 and EG-D$(1+1)$- 10 for budget 10, and similar variants for budget 20 and 40.

Therefore, we have nine flavors of EvolGan, corresponding to different algorithms and budgets.

Human raters

Human raters, chosen through a snowball principle, were utilized for evaluating the two applications in Tables 2.8 and 2.9. A double-blind graphical user interface was employed for presenting images. The evaluation involved the use of a binary question for ethnicity labeling. To maintain objectivity, a double-blind approach was applied throughout the process. The selected raters participated voluntarily, and their motivation ensured stable results. Additionally, the graphical user interface prevented potential biases in the evaluation.

2.2.3 Diversity loss in generative modeling

Usually, modeling decreases the frequency of rare classes.

With StyleGan2, we get 71.55% white people and 4.64% black, according to R (close to [SJJ19]). EvolGan, which is built on top of StyleGan2 with a

Name	Notation	Domain	Note
Variables to be protected			
R	R	$\{A, B, C, D\}$	Ethnicity [Ana19]
AvA	AvA	$\{F, E\}$	Aesthetics[HGS19]
Related auxiliary variables			
	R'	R^4	Logits of R
Koncept512	$K512$	$\mathbb{R}$	IQA
Unrelated auxiliary variables			
Emotions	E	$\{1, 2, 3, 4, 5, 6, 7\}$	facial expression in [Ana19]
	E'	R^{100}	final layer of E
VGG-Face final layer	VF	$\{0, 1\}^{128}$	Binarized VGG-face

Table 2.2: Feature extractors used in the present work. All data are faces, typically generated by StyleGAN2 or other methods in Section 2.2.2.

budget of 40 decreases the percentage of black people to 0% while increasing the frequency of white to 81.25%.

2.2.4 Measuring the diversity loss

We assume that there exist target frequencies for each sensitive class. In the present work, we focus on preserving the diversity in the sense of "having the same frequencies as the frequencies in the original data used for creating the model", so the target frequencies are the frequencies in the original dataset. If we consider the diversity loss associated with optimizing a model, such as EvolGan, we assume that target frequencies are those of the original model.

Given classes $\{1, \dots, n\}$ with target frequencies f_i ($\sum_{i=1}^n f_i = 1$), and real frequencies $f'_1, \dots, f'_n$: the diversity loss Δ is defined as $\Delta := 1 - \inf_{f_i>0} f'_i/f_i$. $\Delta = 0$ if the target frequencies are reached, and $\Delta = 1$ if one of the classes has disappeared.

Throughout our work, we consider diversity loss for classes, and not inside each class: this other important case is left as further work.

2.2.5 Feature extractors

We use various feature extractors (Table 2.2). E and R use VGG-Face [PVZ15]. The goal of these feature extractors is to have auxiliary classes for reweighting: these values, after discretization, provide classes. These classes, termed strata, are used in Section 2.3.2.

2.3 Methods

Section 2.3.1 presents a simple rejection method for ensuring target probabilities in generative modeling.

Section 2.3.2 shows how to build classes in order to apply that method without knowing what the sensitive variables are. Section 2.3.3 then presents a methodology based on multi-objective optimization for improving fairness.

2.3.1 Reweighting: stratified rejection

Consider a generative model on some domain D. Consider a partition $D_1, \ldots, D_m$ of D into m disjoint strata. Assume that some unknown random variable ω has probability $p_i = P(\omega \in D_i)$ and $\sum p_i = 1$. We have another random variable g also living with probability one in the union of the D_i. Assuming that $P(g \in D_i) = p'_i$, a simple tool for building g' such that $P(g' \in D_i) = p_i$ is rejection (see Alg. 1). This simple algorithm generates $g' \in D_i$ with probability p_i.

Algorithm 1: Given a generative model g, bins $D_1, \ldots, D_m$ and their target probabilities $p_1, \ldots, p_m$. This algorithm assumes that none of the D_i has probability 0 for the original generative model g.

Generate x a (new, independent) output of g // random gen

Find i such that $x \in D_i$.

With probability $1 - \dfrac{1}{\max_j \dfrac{p_j}{p'_j}} \dfrac{p_i}{p'_i}$, go back to random gen.

return $g' = x$

2.3.2 Creating strata: reweighting without knowing the target classes

We have classes corresponding to sensitive classes. We consider four sensitive classes of faces (A, B, C, D) using R [Ana19] and two classes using AvA [HGS19] (class F = bottom 20% of the aesthetics variable). However, we also want (possibly non-sensitive) classes used as auxiliary classes for reweighting: our goal is for our method to work for unknown target classes, so we need auxiliary classes. The idea is to investigate how much we can improve fairness for variables A, B, C, D without using those classes in our algorithm. Our auxiliary classes (Section 2.2.5), unrelated to our sensitive classes, will be called strata in the present work: the strata are the D_i used in our reweighting algorithms.

The key point in our experiments "preserving the diversity of unknown target variables" is that we do not use the target variables in our algorithms: our method is unsupervised in this sense.

When we try to maintain diversity for class F, we can use auxiliary variables which are unrelated to F: so, we can use A, B, C and D. And when we try to maintain diversity for classes A, B, C and D, we can use F as an auxiliary variable.

Some attributes (final layer of an emotion classifier, or technical quality of the photo) can be used for all classes as they are not directly related to any of our sensitive variables. We will use two parameters d and M in our experiments. Given a possibly large number of auxiliary variables (not the target variables), we select d variables. Each of these d variables is discretized in M values, where M is called the arity: thresholds are chosen so that the M values are equally frequent.

2.3.3 The user-assisted context: generating multiple solutions

Whereas in Section 2.3.2 we have considered a drop-in replacement of the baseline, which generates one image per instance, we now consider the case in which we generated several instances, and the user can select one of them (see Alg. 2). There are two parts:

- how to generate multiple contexts, and,
- for some methods which generate way too many solutions for being manually searched by a human user, how to sample the obtained Pareto front.

2.3.3.1 How to Generate Multiple Solutions

We consider a fixed limit on the number of generated images allowed so that the tool remains manageable for the user. Several approaches can generate a targeted number of outputs; we consider

- multi-objective optimization (MOO: splitting the original criterion into several and optimizing them jointly) and
- multiple runs.

Doing multiple runs is a simple and intuitive solution for generating multiple images. Regarding MOO, our solution is not compatible with all generative models: we consider that images are obtained by numerical optimization of a linear combination of criteria [RTR+19]. Instead of aggregating them, [CRRN+21] proposed to preserve diversity by optimizing

Algorithm 2: Image generation

No context, no user assistance	**Context, no user assistance**	**Context, user assistance**
Repeatedly, generate one individual per request.	Repeatedly, generate one individual per request. Requests have a context (e.g., low-resolution image).	Repeatedly, generate k individuals per request (e.g., by Pareto-based MOO, or by diversity-based MOO, or by MSR): the user chooses one of them.
Check that their frequencies match the expectation: compute a DL.	Check that their frequencies match the frequencies of the context (e.g., same ethnicity as low-res image): compute a DL.	Check that their frequencies match the contextual expectations: compute DL.

Different contexts for image generation, without or with human assistance.
Left: unassisted context, generative model.
Middle: generative model with target class (case in which there is an expected class, e.g., super-resolution in which the ethnicity is supposed to be preserved statistically).
Right: user-assisted method.
Not all unsupervised fairness methods can be applied in all cases.
The reweighting method in Sections 2.3.1 and 2.3.2 can be applied to the two first columns. In contrast, the multiple generation such as the one in Section 2.3.3 can be applied to the third column only.

several numerical criteria by MOO, and we include this technique (as well as the previously mentioned reweighting techniques) in our fairness context. MOO naturally generates several solutions instead of one so that we are (presumably) more likely to have at least one satisfactory solution.

2.3.3.2 How to sample the obtained solutions

When we do multiple runs, we can choose their number to control the number of generated images. However, in MOO, we typically get a Pareto front. This Pareto front might be huge. Therefore, we have to sample this Pareto front. There are many tools for this:

- Optimizing this sampling for some representativeness criterion in the fitness space (hypervolume and others).
- Or maximizing some diversity criterion in the original domain, regardless of fitness values.

2.4 Methods analysis

2.4.1 Multi-objective diversification

Generating several solutions and letting the user choose among those proposals is a simple workaround for partially mitigating diversity loss.

However, not all methods are equal: we would like to have as much diversity as possible for a given fixed number of proposals. Also, Fig. 2.1 shows that it is not obvious that this will work: though this might not be intuitive, one can design counter-examples in which focusing on the Pareto-front and even more on a few key elements representing the Pareto front can actually decrease the diversity, compared to generating just one image at a time, because the Pareto frontier might be entirely covered by a single class (in particular the biggest class, for which values are usually greater in machine learning models, as explained in Section 2.2.1).

The simplest, and maybe most robust solution is to run multiple independent (randomized) runs: if the probability $P(g \in C)$ of generating a point in C is low, then the probability $1 - (1 - P(g \in C))^k$ of having at least one of k generated image inside C is greater:

$$1 - (1 - P(g \in C))^k \geq P(g \in C) (\text{strict if} P(g \in C) \notin \{0, 1\}).$$

If the user needs an image of class C, generating k images is more likely to have at least one in C unless the original probability is 0 or 1.

The question is now how to do better than this baseline. We consider the following ideas:

- the k runs are not using the same weights: e.g., we use random weights in the optimization runs, and they are randomly drawn at each run.

- we run a MOO algorithm which tries to maximize some quantity, e.g., the hypervolume of the obtained solutions, or their diversity in the loss space, or the coverage in the domain space.

Consistent with the credo of the present work (not using target classes in the algorithm), these algorithms are independent of the target classes.

2.4.2 Stratification by rejection is rarely detrimental

The reweighting method in Section 2.3.1 works in the sense that, by design, when we use it, we switch back to the exact probabilities for each stratum, i.e., $p' = p$. This implies that, unless a target class has entirely disappeared in the model, reweighting using strata based on the target classes recovers the frequencies of all target classes. However, the point of the present work is to fix frequencies of unknown target classes.

So, now, consider a target class C, which is not necessarily one of the strata. If C is one of the D_i (or a union of them) then, as discussed above, the stratification leads to $p(g \in C) = p(\omega \in C)$: let us see if we can find a more general case in which $P(g \in C) = P(\omega \in C)$.

The Diversity Loss (DL) measure we are using (Section 2.2.4) for estimating the DL of a model g compared to a random variable w is based on aggregating measures of DL for several classes: the global diversity loss is $\Delta := 1 - \inf_{f_i>0} f'_i/f_i$ where f_i is the target frequency for class i and f'_i is the observed frequency.

$$\begin{aligned}
\Delta &= \max_{C;P(w\in C)>0} \left(1 - \frac{P(g \in C)}{P(w \in C)}\right) \\
&= \max_{C;P(w\in C)>0} \frac{1}{P(w \in C)} \left(P(w \in C) - P(g \in C)\right) \\
&= \max_{C;P(w\in C)>0} \frac{1}{P(w \in C)} \left(pq - pq'\right)
\end{aligned}$$

where:

- q_j is the probability of class C in stratum D_j for the original random variable w i.e. $q_j = P(w \in C | w \in D_j)$;

- q'_j is the counterpart for the model g i.e. $q'_j = P(g \in C | g \in D_j)$.

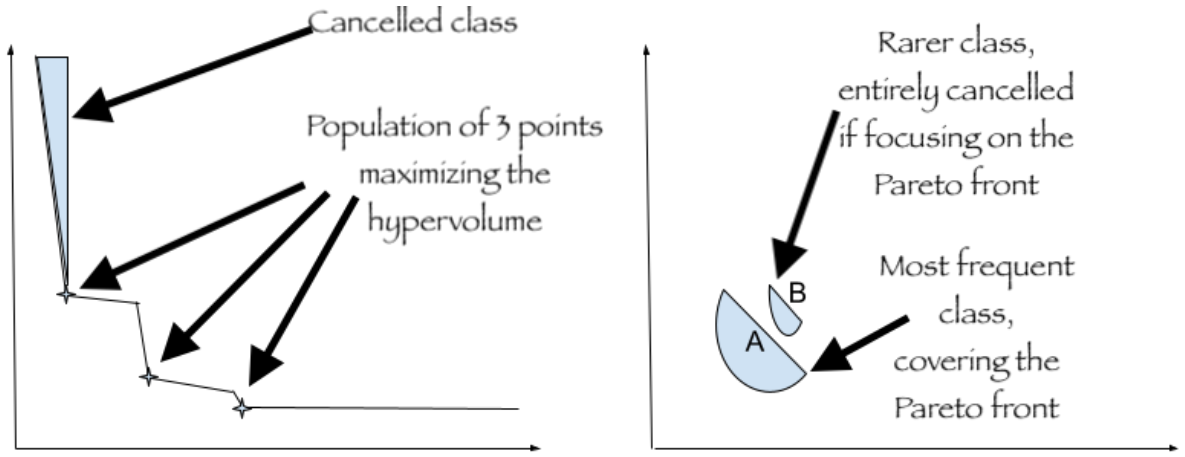

Figure 2.1: Bi-objective minimization, cases in which Pareto-dominance will be detrimental to diversity.
Left: artificial counter-example showing that maximizing a numerical diversity criterion (the hypervolume) over the Pareto front might not provide diverse solutions. Here, we see a Pareto-front and the hypervolume-best approximation by 3 points.
Dots: the 3 individuals maximizing the hypervolume.
Gray areas: examples of classes that completely disappear if we consider those dots (as they maximize the hypervolume) rather than a random sampling of the Pareto front.
Right: other counter-example. Class A is assumed to be much bigger than class B, and to have, therefore, better scores for both criteria: this is because, as discussed in the text, bigger classes typically have better scores (see Section 1.5.2).
While local optimization from points in B will provide points in B, a global optimization based on Pareto fronts will provide only points in A: class B is not represented.

The reweighting increases the DL for class C if $pq - pq' > pq - p'q'$ (where pq is short for $\sum_j p_j q_j$). This is equivalent to $q'(p' - p) > 0$ and $p(q - q') > 0$. This means that reweighting is detrimental for this measure if two conditions occur simultaneously:

1. $p(q - q') > 0$
2. $q'(p' - p) > 0$,

where the condition (1.) means that $q - q'$ is overall positive on average for the frequencies p (i.e., g tends to underestimate class C), which is precisely the case of interest: this means that g is not doing well on C. And the condition (2.) $q'(p' - p) > 0$ implies that we tend to overestimate classes in which C has a low probability, which contradicts the general assumption "diversity loss usually occurs for rarer classes" in Section 2.2.3. Therefore, it seems unlikely that reweighting can worsen diversity loss, at least for this measure.

2.5 Experimental results

2.5.1 Framework

We compare our methods in different contexts. Each context (g, b) is defined by a generative model g to be compared to a baseline b (dataset or model). We check if g has a diversity loss, comparatively to b. We have 18 contexts, as described below.

The baseline b is either a dataset or a PGAN [KALL18b] trained on it (i.e., two possibilities here), and we try to fix the diversity loss when applying EvolGan [RTH+20] with budget 10, 20, 40 (3 possibilities) and algorithm DOPO [RTR+19], CMA [HO03] or random search (3 possibilities): g can be any of these 9 combinations, and we consider the diversity loss compared to one of the two different possible b, hence 18 contexts (Table 2.3).

Different contexts have different diversity losses: typically, CMA or RandomSearch lead to more diversity loss than DOPO.

We have checked that (naively) optimizing technical quality is detrimental to fairness (Section 2.5.2).

We show (Section 2.5.3) that applying reweighting according to target classes is unsurprisingly more effective than reweighting according to unrelated strata, but the latter methodology still does mitigate fairness issues.

Then Section 2.5.4 compares various forms of user-assisted optimization for tackling fairness issues.

EG variant	Diversity loss	Remaining diversity loss (%)
EG-CMA-10	0.675	97.587
EG-CMA-20	0.778	96.505
EG-CMA-40	0.872	90.098
EG-D(1+1)-10	0.108	70.592
EG-D(1+1)-20	0.204	94.112
EG-D(1+1)-40	0.333	88.999
EG-RandomSearch-10	0.675	87.333
EG-RandomSearch-20	0.785	88.270
EG-RandomSearch-40	0.876	96.438

Table 2.3: Diversity loss for class F (i.e., low aesthetics value according to AvA) for EG compared to PytorchGanZoo (EG is an improvement of PytorchGanZoo using K512 as an IQA for biasing the latent vaariables). The diversity loss depends on how strongly we improve the GAN using EvolGan (more budget = more improvement in terms of quality measured by K512).
We also show (third column) how much the diversity loss is preserved in spite of reweighting w.r.t. E: numbers $< 100\%$ show that a part of the diversity loss is repaired. No number is greater than 100%: our method is never detrimental.

2.5.2 (Naively) optimizing → less diversity

We train a PGAN [KALL18b] and then improve it using IQA as in [RTH+20]: PGAN → EG10 → EG20 → EG40 (each "→" being an improvement in terms of image quality by refining the latent variables using the image quality assessment tool as a criterion[RTH+20]).

As noted in [RTH+20], the quality improvement in EvolGAN is related to some diversity losses: for horses, we get rid of bugs such as horses with 3 heads, which is in some sense a sort of diversity loss. Unfortunately, this also reduces diversity in the sense that relevant rare classes become rarer (Table 2.3): there is a diversity loss from the dataset to the PGAN, and this diversity loss is increased when we increase the budget of the GAN improvement by EvolGan.

Baseline	Model	M	Diversity loss before reweight	Percentage of DL remaining with $d = 2$	Percentage of DL remaining with $d = 4$
PGAN	EG-CMA-10	3	0.442	53.266	42.257
PGAN	EG-CMA-20	3	0.513	49.901	32.176
PGAN	EG-CMA-40	3	0.663	83.654	40.683
PGAN	EG-D(1+1)-10	3	0.080	74.254	16.168
PGAN	EG-D(1+1)-20	3	0.070	72.913	30.008
PGAN	EG-D(1+1)-40	3	0.115	25.079	33.147
dataset	EG-RandomSearch-0	3	0.314	31.699	25.398
dataset	EG-RandomSearch-10	3	0.563	28.083	33.860
dataset	EG-RandomSearch-20	3	0.644	33.280	40.709
dataset	EG-RandomSearch-40	3	0.738	65.564	63.747
dataset	EG-CMA-0	3	0.343	40.314	16.914
dataset	EG-CMA-10	3	0.561	32.505	6.927
dataset	EG-CMA-20	3	0.617	27.205	29.584
dataset	EG-CMA-40	3	0.735	47.673	33.604
dataset	EG-D(1+1)-0	3	0.312	40.628	10.630
dataset	EG-D(1+1)-10	3	0.339	32.440	28.977
dataset	EG-D(1+1)-20	3	0.347	95.618	11.370
dataset	EG-D(1+1)-40	3	0.350	32.822	16.938

Table 2.4: Impact of reweighting with related variables on the diversity loss for classes A, B, C, D: we see that the original diversity loss is significant (4th column) and reduced a lot if we use 4 variables for reweighting (6th column). Even 2 variables contribute quite well to a significant reduction of diversity loss (5th column). Dataset: faces generated by StyleGAN2. Strata used for reweighting: logits of the output layer of R discretized with $M = 3$ and $d = 2$ (5th column) or $d = 4$ (6th column).

2.5.3 Reweighting mitigates fairness issues

2.5.3.1 Classes A, B, C, D

Table 2.4 presents the diversity loss and the fixed diversity loss when using reweighting. We use 2 or 4 variables correlated (though not equal) to the target attribute, namely the discretized predicted probabilities of the 4 modalities of the target class. As variables are correlated to the target problem, results are excellent.

We now switch to a more challenging case. Table 2.5 compares various discretizations in the difficult context of reweighting variables unrelated to the target variables. For example, (80,8) means that we use $d = 80$ variables and split each of them in $M = 8$ bins. We got the best results with 10 variables discretized in 3.

There are four target classes for faces unrelated to emotions. The variables are the final layer of an emotion recognition network.

Still, in that difficult case, Fig. 2.2 shows how diversity losses are moved in the right direction by the reweighting – not much, but beneficial, and most importantly, not detrimental.

2.5.3.2 Class E: confirming results for reweighting with unrelated variables

Table 2.6 and Table 2.7 present the impact of reweighting using the probabilities of class A, B, C and D (discretized) on the diversity loss of class E. (ABCD) and E are unrelated, so this is unsupervised fairness improvement.

2.5.4 Multi-objective optimization: only some forms of MOO mitigate fairness issues

MOO typically has two phases:

- optimization run, building a possibly large Pareto front;
- selection of a reduced Pareto front for presentation to the user.

This does not cover all MOO methods. The second stage is not always present, as some tools are equipped with a mechanism for navigating the Pareto front. Also, sometimes the first stage includes inputs from the human. We will nonetheless consider the framework above in the present work.

As mentioned before, a simple solution for MOO is to do multiple simple runs (MSR): just run the algorithm several times, and consider the several outputs. We consider other methods, namely maximizing the hypervolume

Number of vars d	Discretization M	DL before reweighting	DL after reweighting
1	2	0.431	0.421
1	3	0.431	0.428
1	5	0.431	0.435
1	8	0.431	0.430
2	3	0.431	0.403
2	5	0.431	0.431
2	8	0.431	0.433
4	2	0.431	0.414
4	3	0.431	0.403
4	5	0.431	0.428
4	8	0.431	0.427
10	3	0.431	0.395
10	5	0.431	0.423
20	2	0.431	0.419
20	3	0.431	0.401
20	5	0.431	0.432
20	8	0.431	0.428
80	2	0.431	0.419
80	8	0.431	0.428

Table 2.5: Diversity loss for (A, B, C, D) after reweighting, in our hardest context (variables very uncorrelated to the target variable, namely E'). We observe that in most cases, the reweighting is still beneficial compared to 0.431 originally, though this difficult case does not lead to drastic improvements.
Dataset: faces generated by StyleGan2.
Strata: discretization of E' with $d \in \{1, 2, 4, 10, 20, 80\}$ and $M \in \{2, 3, 5, 8\}$.

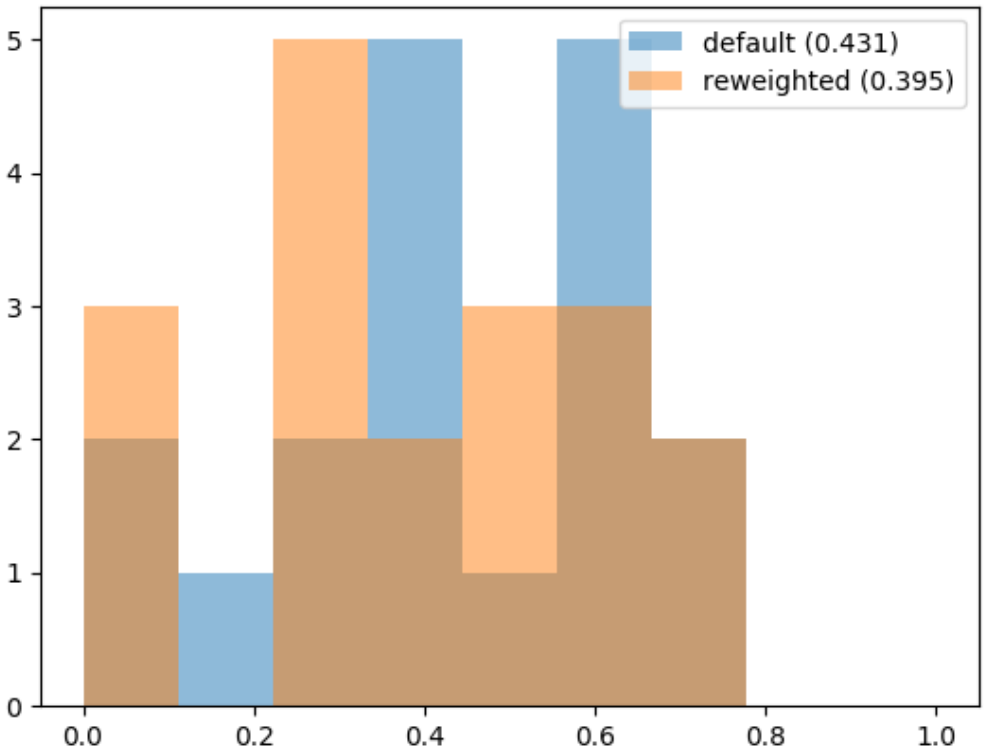

Figure 2.2: Hard case with unrelated reweighting variables: histogram of diversity losses for (A,B,C,D) using reweighting based on strata of R', over each of 18 contexts (see text). The method is slightly beneficial; the average moves from 0.431 to 0.395. We use the best method in Table 2.5, rerun from scratch for mitigating the hyperparameter selection bias, getting the same 0.395 value.
Dataset, strata, as in Table 2.5.
X-axis: DL.
Y-axis: number of contexts (out of 18) with DL falling in the given DL bin.

Source	Target	d	M	DL	Remaining DL(%)
PGAN	EG-CMA 10	1	8	0.675	99.832
PGAN	EG-CMA 20	1	8	0.778	100.444
PGAN	EG-CMA 40	1	8	0.872	100.923
PGAN	EG-D(1+1) 10	1	8	0.108	103.808
PGAN	EG-D(1+1) 20	1	8	0.204	76.961
PGAN	EG-D(1+1) 40	1	8	0.333	92.000
PGAN	EG-RandomSearch 10	1	8	0.675	88.797
PGAN	EG-RandomSearch 20	1	8	0.785	100.171
PGAN	EG-RandomSearch 40	1	8	0.876	98.348
PGAN	EG-CMA-10	2	8	0.675	96.660
PGAN	EG-CMA-20	2	8	0.778	95.140
PGAN	EG-CMA-40	2	8	0.872	89.906
PGAN	EG-D(1+1)-10	2	8	0.108	103.808
PGAN	EG-D(1+1)-20	2	8	0.204	89.274
PGAN	EG-D(1+1)-40	2	8	0.333	98.751
PGAN	EG-RandomSearch-10	2	8	0.675	87.870
PGAN	EG-RandomSearch-20	2	8	0.785	91.072
PGAN	EG-RandomSearch-40	2	8	0.876	97.075

Table 2.6: Impact of reweighting on diversity loss for class E when using classes R as auxiliary variable. Part 1. We see that adding variables almost always improves results, and cases in which reweighting is detrimental are rare. Dataset: faces generated by StyleGan2. Sensitive variables for which DL is computed: emotions. Strata: IQA values provided by R', i.e., logits of R, with discretization with $d \in \{1, 2, 3, 4\}$ variables and $M = 8$ equally likely bins per variable.
Observation: increasing d reduces the DL after reweighting.

Source	Target	d	M	DL	Remaining DL(%)
PGAN	EG-CMA-10	3	8	0.675	95.538
PGAN	EG-CMA-20	3	8	0.778	95.218
PGAN	EG-CMA-40	3	8	0.872	89.552
PGAN	EG-D(1+1)-10	3	8	0.108	90.259
PGAN	EG-D(1+1)-20	3	8	0.204	81.139
PGAN	EG-D(1+1)-40	3	8	0.333	84.623
PGAN	EG-RandomSearch-10	3	8	0.675	86.845
PGAN	EG-RandomSearch-20	3	8	0.785	87.502
PGAN	EG-RandomSearch-40	3	8	0.876	95.705
PGAN	EG-CMA-10	4	8	0.675	97.587
PGAN	EG-CMA-20	4	8	0.778	96.505
PGAN	EG-CMA-40	4	8	0.872	90.098
PGAN	EG-D(1+1)-10	4	8	0.108	70.592
PGAN	EG-D(1+1)-20	4	8	0.204	94.112
PGAN	EG-D(1+1)-40	4	8	0.333	88.999
PGAN	EG-RandomSearch-10	4	8	0.675	87.333
PGAN	EG-RandomSearch-20	4	8	0.785	88.270
PGAN	EG-RandomSearch-40	4	8	0.876	96.438

Table 2.7: Impact of reweighting on diversity loss for class E when using classes R as auxiliary variable. Part 2. We see that adding variables almost always improves results, and cases in which reweighting is detrimental are rare. Dataset: faces generated by StyleGan2. Sensitive variables for which DL is computed: emotions. Strata: IQA values provided by R', i.e., logits of R, with discretization with $d \in \{1, 2, 3, 4\}$ variables and $M = 8$ equally likely bins per variable.
Observation: increasing d reduces the DL after reweighting.

for phase 1 and using various techniques such as IGD, EPS, RANDOM for constructing a subset.

In particular, to extract $1 \leq m \leq n$ points from an approximate Pareto set $\{x_1, \ldots, x_n\}$, a range of approaches can be used:

- Random subset:

  just pick up m of the x_i, uniformly at random and without replacement.

- HV:

  pick up $\{x_{j_1}, \ldots, x_{j_m}\}$ such that their Hypervolume C_h is maximal.

- Loss-covering, also known as IGD (inverted generational distance, [SAT04]):

  pick up $\{x_{j_1}, \ldots, x_{j_m}\}$ such that

  $$C_l = \sum_{i=1}^{n} \inf_{j \leq m} ||F(x_i) - F(x_{i_j})||^2$$

  is minimal, where $F(x) = (f_1(x), \ldots, f_N(x))$.

- COV (covering the Pareto-front):

  pick up $\{x_{j_1}, \ldots, x_{j_m}\}$ such that

  $$C_d = \sum_{i=1}^{n} \inf_{j \leq m} ||x_i - x_{i_j}||^2$$

  is minimal.

- Additive epsilon approximation (EPS, [PY00]):

  pick up $\{x_{j_1}, \ldots, x_{j_m}\}$ such that

  $$C_e = \max_{i=1}^{n} \inf_{j \leq m} ||F(x_i) - F(x_{i_j})||_\infty$$

  is minimal, where

  $$F(x) = (f_1(x), \ldots, f_N(x))$$

  .

In domain-covering, we do the same as covering the Pareto-front (COV), but over all generated points and not only the Pareto-front.

Tables 2.8 (target class is black) and 2.9 (target class is female Asian) show that the best results concerning maximum diversity are obtained by

Algorithm	Selector	Percentage
9 single-objective runs		
NGOpt 9	domain-covering	**33**
NGOpt 9	eps	**33**
NGOpt 9	loss-covering	**33**
NGOpt 9	msr	**33**
CMA		
CMA	domain-covering	33
CMA	eps	33
CMA	loss-covering	44
CMA	**msr**	**66**
Portfolio Discrete-$(1+1)$		
PortfolioDiscrete$(1+1)$	msr	16
PortfolioDiscrete$(1+1)$	eps	33
PortfolioDiscrete$(1+1)$	loss-covering	33
PortfolioDiscrete$(1+1)$	**domain-covering**	**83**
Differential Evolution		
DE	loss-covering	16
DE	eps	16
DE	domain-covering	33
DE	**msr**	**55**
Random Search		
RandomSearch	loss-covering	0
RandomSearch	msr	33
RandomSearch	eps	50
RandomSearch	**domain-covering**	**66**

Table 2.8: Multi-objective inspirational generation: the target is the face of a black person, originally very pixelized; the goal is to approximate it with PytorchGanZoo. We consider with which probability PytorchGanZoo generates at least one face of the correct ethnicity. Each algorithm generates nine faces.

The best selector consists of picking up the nine outcomes of nine single runs (MSR: multiple single runs) or using domain covering, i.e., never using a Pareto-based measure.

In conclusion, multi-objective optimization does work for generating diversity. However, we should not use Pareto-dominance and focus on multiple outcomes of random single-objective runs or diversity in the domain ("domain-covering" method), because fitness-based measures are too biased for being used for diversity.

Algorithm	Selector	Percentage
9 single-objective runs		
NGOpt 9	**domain-covering**	**22**
NGOpt 9	eps	0
NGOpt 9	loss-covering	5
NGOpt 9	msr	11
CMA		
CMA	**domain-covering**	**27**
CMA	eps	11
CMA	loss-covering	22
CMA	msr	0
Portfolio Discrete-$(1+1)$		
PortfolioDiscrete$(1+1)$	msr	0
PortfolioDiscrete$(1+1)$	eps	38
PortfolioDiscrete$(1+1)$	loss-covering	16
PortfolioDiscrete$(1+1)$	domain-covering	38
Differential Evolution		
DE	loss-covering	16
DE	**eps**	**22**
DE	domain-covering	5
DE	msr	11
Random Search		
RandomSearch	loss-covering	5
RandomSearch	msr	0
RandomSearch	eps	11
RandomSearch	**domain-covering**	**33**

Table 2.9: Counterpart of Table 2.8 for female Asian target. As in Table 2.8, domain-covering performs best.

domain-covering or by MSR, and not by MOO approaches focusing on diversity over the Pareto front. The effective diversity measures are not based on Pareto-dominance. The best results are obtained either by pure MSR, using multiple runs and keeping all results, or by domain-covering, i.e., creating a subset using diversity in the image domain.

This result is not so intuitive, so we ran additional experiments to check if Pareto-dominance can be detrimental to diversity.

We conclude that Pareto-based MOO can be detrimental to diversity even with a large budget and 16 generations instead of 1.

This is shown by Table 2.10: we do an additional experiment based on Pytorch-Gan-ZOO and variants. We use both single-objective optimization (EvolGan with budget 10000) and our MOO counterpart. We get a single image per run for single-objective optimization, and we can estimate DL as usual. We use MOO, with three objectives linearly combined in the single-objective case: minimizing the squared of the injected latent variables, maximizing the IQA score, and maximizing the discriminator score. We use a large budget and many generated individuals so that problems can not be attributed to the parametrization. We consider that the "frequency" of a class is the frequency at which at least one of the outputs contains that class (see Alg. 2).

We see that MOO by classical Pareto-dominance is not always solving diversity issues. It works only when the method has over-optimized and completely destroyed diversity (Table 2.10: results are $< 100\%$ in the last column only if the diversity loss is $> 95\%$). Whereas diversity in the domain (domain-covering) or simple multiplication of runs (as in MSR) works in many cases, optimization with Pareto-dominance can fail.

We conclude that counter-examples as in Fig. 2.1 are not an exception but the standard behavior of Pareto-dominance: due to different scales of quality depending on the frequency of classes, we can not reliably use Pareto-dominance for selecting samples. MSR is the only method that did not have counter-examples. MOO methods based on Pareto fronts were ok only when the method for extracting representative images was based on domain-covering, i.e., unsupervised correction.

2.6 Conclusion

Here is the list of the key findings of this chapter:

- **Quality improvement degrades diversity:**

Original model	EG40 variant	PF size	Subset	d, M	Diversity loss	Uncancelled loss (%)
PGAN	EG-RandomSearch	16	COV	5,2	0.726	111.333
PGAN	EG-CMA	16	COV	5,2	0.977	89.285
PGAN	EG-D(1+1)	16	COV	5,2	**0.707**	116.297
PGAN	EG-RandomSearch	16	IGD	5,2	**0.72**	112.165
PGAN	EG-CMA	16	IGD	5,2	0.973	90.008
PGAN	EG-D(1+1)	16	IGD	5,2	0.730	112.409
PGAN	EG-RandomSearch	16	Random	5,2	**0.697**	118.724
PGAN	EG-CMA	16	Random	5,2	0.969	89.622
PGAN	EG-D(1+1)	16	Random	5,2	0.726	115.778
PGAN	EG-RandomSearch	16	EPS	5,2	0.738	107.916
PGAN	EG-CMA	16	EPS	5,2	0.977	88.095
PGAN	EG-D(1+1)	16	EPS	5,2	**0.709**	116.377

Table 2.10: Column 6 shows the DL when moving from the original (column 1) to the improved version (column 2), and column 7 presents the part of this DL which is not solved by applying MOO for generating 16 points.
There is a strong computational budget (10000) and a large generated set (16 points) in the present context.
We consider that the result is ok if at least one of those 16 generations is of the expected class.
Column 7 is frequently above 100%, i.e., results are **worse** than in the single-objective case generating only one image: this shows that even with favorable conditions, MOO based on Pareto-dominance can be detrimental.
Only MSR (running several times and gathering the results) or domain-covering (i.e., good diversity for a side measure in the domain) provide stable improvements in the user-assisted context (Tables 2.8 & 2.9).
Dataset: CelebaHQ (see https://github.com/tkarras/progressive_growing_of_gans).
Model: PytorchGanZoo.
Method: described in Section 2.3.3.
Sensitive variables on which DL is measured: ethnicity.

We checked that improving the visual quality degrades diversity when biasing latent variables through IQA methods. The biasing effect is consistent with known facts.

To mitigate this issue, we propose two methods.

  - The first (Alg. 1) is a drop-in improvement of a generative model: it can be applied as soon as we have some auxiliary features that we can use for defining strata.
  - The second one is user-assisted (Alg. 2) and can use MOO (either with Pareto-dominance for selecting a subset or with diversity preservation for some features in the domain) or MSR.

- **Reweighting by related auxiliary variables:**

  Unsurprisingly, reweighting by auxiliary variables close to the target classes is very effective at reducing the diversity loss. We cancel the diversity loss when reweighting using the same target class. This incurs a computational cost and does not solve quality inside each class, but we recover target frequencies.

- **Reweighting by unrelated auxiliary variables:**

  A good finding is that we never degrade performance by applying reweighting, even when using unrelated variables. There are good reasons for this (Section 2.4.2). We recommend reweighting by as many variables as possible (at least as long as there is data enough for computing statistics with enough precision). However, we acknowledge that this has a computational cost.

- **Using MOO, also without knowing categories:**

  The idea of using MOO for generating diversity is intuitively appealing. The only multi-objective method which was never detrimental to diversity loss is MSR, i.e. simply running several times (with randomly drawn linear combination coefficients) the single-objective method and proposing the obtained solutions.

  Domain-covering, which is an unsupervised selection as it uses auxiliary variables only (and not the target variables), was also satisfactory and sometimes the best.

  MOO with Pareto-dominance for selecting a final subset presented to the user is risky: in spite of the intuition “MOO increases diversity”,

we did increase the diversity loss when using Pareto-dominance because the quality measures have different ranges for different classes. Effects as in Fig. 2.1 turn out to be a real issue when using MOO for diversity.

Side remarks & caveats

- **Combination with supervised fairness:**

  we considered purely unsupervised fairness, but we could do the same in combination with given sensitive variables: after a first correction for given sensitive variables, we can add a correction with respect to some unrelated generic strata.

- **Impact of the optimization method:**

  Tables 2.3, 2.4 and 2.10 show that CMA leads to more diversity loss compared to random search or PortfolioDiscrete(1+1). This is reasonable as the prior distribution is ignored by CMA, whereas it impacts every other tested methods:

  - Random search uses the prior distribution at each step for choosing a point;
  - Discrete $(1 + 1)$ algorithms use the marginal of the probability distribution for each modified variable.

We presented results for reweighting with statistics based on large datasets, so that there was no problem for precisely estimating p_i/p_j as needed: with small datasets, precision might be an issue.

3

Enhancing Image Diversity in Text-to-Image Generation

3.1 Introduction

Figure 3.1: Images generated with standard Stable Diffusion and our method for the prompt "rose", batch size = 5. Here, our method is shown to select images not only diverse in colors, but also in ideas, compared to Stable Diffusion.

In this chapter, we present Diverse Diffusion, *i.e.*, modifications to the Stable Diffusion algorithm that facilitate the generation of diverse images and thereby help to create more inclusive art within a limited number of

Stable Diffusion generations (and therefore less computational power) in unsupervised fashion. Images generated with and without our approach are illustrated in Figure 3.1.

We highlight the following contributions:

- A general unsupervised technique that can be applied to existing text-to-image models to increase image diversity, which is essential for generating realistic and varied images.
- Experiments that demonstrate the diversity advantages of our proposed approach.

3.2 Diversity algorithms

There are different approaches to generating diverse point sets in an unsupervised manner. For example, Latin Hypercube Sampling [MBC79], low discrepancy methods [Nie92, Ham60, Ata04] and low dispersion. Here, we focus on low dispersion, optimized by a very limited random search for staying in the domain of validity of our latent variables. We aim to find vectors in the latent space of Stable Diffusion that are distant from one another. To accomplish this, we generate multiple vectors in the latent space until we obtain a set that contains the required number of vectors (determined by the batch size) and satisfies a specific distance requirement.

In the "**baseline**" setting, we generate images using the standard, unmodified version of Stable Diffusion without imposing any distance requirements on the latent space.

In the "**cap**" setting, we enforce a minimum requirement of $d_{\min}$ on the vectors corresponding to all pairs of images within a batch. We illustrate the procedure of choosing a set of latent vectors V for batch size B and minimum distance requirement d_{min} in algorithm 3.

In the **"max"** setting, we impose a maximum number of iterations on searching for a new vector that would have a maximal minimal distance to all the already selected vectors in the batch. We illustrate the procedure of choosing a set of latent vectors V for batch size B and a maximum number of iterations requirement $N_{\max}$ in algorithm 4.

In setting **"pooling_cap"** and **"pooling_max"** we apply the same exact methods as in "cap" and "max" but the distance is calculated differently. Specifically, the distance is computed between the vectors that were initially processed by average pooling 8×8 which down-samples the vector size to $4 \times 8 \times 8$.

Algorithm 3: Generating diverse vectors in the "cap" setting

Require: batch size B, minimum distance d_{min}
Ensure: Set of diverse vectors V
  $V \leftarrow \emptyset$ {Initialize an empty set of vectors}
  while $|V| < B$ **do**
    $v_{\text{new}} \leftarrow$ GenerateNewVector() {Generate a new vector in the latent space of stable diffusion}
    if MinDistance$(v_{\text{new}}, V) \geq d_{\text{min}}$ **then**
      $V \leftarrow V \cup \{v_{\text{new}}\}$ {Add the new vector to the set}
    end if
  end while
  return V {Return the set of diverse vectors}

We use two different settings for generating diverse image batches: **standard experiment** and **long experiment**.

The parameters of a standard experiment are the following:

- In the setting "cap" the minimal distance between latent vectors should be at least 182
- In the setting "pooling_cap" the minimal distance between latent vectors (after pooling operation) should be at least 3.1
- In the setting "max" and "pooling_max" the number of iterations after which the farthest vector is found is 100.

The parameters of a long experiment are the following:

- In the setting "cap" the minimal distance between latent vectors should be at least 183.
- In the setting "pooling_cap" the minimal distance between latent vectors (after pooling operation) should be at least 3.1.
- In the setting "max" and "pooling_max" the number of iterations after which the farthest vector is found is 10000.

The choice of settings provides variable diversity levels and variable computational complexity. In the current chapter for small batch sizes $(3, 5, 10)$, we create both standard and long experiments, and for the big batch sizes (50) we create only standard experiments. In the cap and pooling cap setting, due to the distance limitations and no limit on a number

Algorithm 4: Generating diverse vectors in the "max" setting

Require: batch size B, maximum iterations number $N_{\max}$
Ensure: Set of diverse vectors V
  $V \leftarrow \emptyset$ {Initialize an empty set of vectors}
  for $b = 1$ **to** B **do**
    $v_{\text{farthest}} \leftarrow$ GenerateNewVector()
    for $N = 1$ **to** $N_{\max} - 1$ **do**
      $v_{\text{new}} \leftarrow$ GenerateNewVector() {Generate a new vector in the latent space of stable diffusion}
      if MinDistance$(v_{\text{new}}, V) >$ MinDistance(v_{farthest}, V) **then**
        $v_{\text{farthest}} \leftarrow v_{\text{new}}$
      end if{Find the vector with the maximum minimum distance to the existing set}
      $V \leftarrow V \cup \{v_{\text{farthest}}\}$ {Add the farthest vector}
    end for
  end for
  return V {Return the set of diverse vectors}

of iterations, the experiment for big batches becomes increasingly slower. That is why, for purposes of limiting computational cost, we recommend the "pooling_max" method for batch sizes more than 50.

3.3 Evaluation Methods

In order to evaluate diversity of generated images, we use both quantitative and human evaluation methods. For quantitative methods, we focus primarily on color diversity image similarity metric LPIPS [ZIE18a]. Increasing image color diversity is an important aspect of ensuring general diversity of the images: we would like to get representation for people, animals, and objects of all colors.

3.3.1 Color diversity

To assess the color diversity in an image batch b, we employ a method that involves extracting color information from each image in the batch using the RGB color model, which represents colors as a combination of red, green, and blue channels. Specifically, we compute the mean value for each channel (R_i, B_i, G_i) in the given image i, and identify whether one of these colors is predominantly present in the image i.

To determine the dominant color in image i with respect to a coefficient

K, we define the image as having a dominant color of

- Red if $R_i > K \times max(G_i, B_i)$,
- Green if $G_i > K \times max(R_i, B_i)$,
- Blue if $B_i > K \times max(G_i, R_i)$, and
- None if none of these inequalities are true.

We denote the dominant color of image i with respect to the coefficient K as $D_K(i)$.

To evaluate color diversity of an image batch b, we compute a number of dominant colors in that batch with respect to a coefficient K, $N_K(b)$.

$$\begin{aligned} N_K(b) = 3 &- \prod_{i \in b}(1 - I(D_K(i) == Red)) \\ &- \prod_{i \in b}(1 - I(D_K(i) == Green)) \\ &- \prod_{i \in b}(1 - I(D_K(i) == Blue)), \end{aligned} \tag{3.1}$$

where I is an indicator function and $D_K(i)$ is a dominant color of an image i with respect to the coefficient K.

The first color diversity metric across the batches set B is an average number of dominant colors present in the batches of that set with respect to a coefficient K.

$$Avg_K(B) = \frac{\sum_{b \in B} N_K(b)}{||B||}, \tag{3.2}$$

where $||B||$ is the total number of batches in the set B.

The second color diversity metric aims to compute proportion of batches that contain at least one image predominantly exhibiting red, blue or green color, considering RGB image encoding. This allows for the quantification of color variability within a batch.

Specifically, for various values of a coefficient K and batch set B we compute the proportion

$$C3_K(B) = \frac{\sum_{b \in B} I(N_K(b)) == 3}{||B||}, \tag{3.3}$$

where $||B||$ is the total number of batches in the set B and I is an indicator function.

Another color diversity metric is the proportion of batches containing images with different dominant colors (not all the images in a batch are predominantly red, blue or green).

Specifically, for various values of a coefficient K and batch b we compute the proportion

$$C2_K(B) = \frac{\sum_{b \in B} I(N_K(b)) >= 2}{||B||}, \tag{3.4}$$

where $||B||$ is the total number of batches in the set B and I is an indicator function.

3.3.2 LPIPS metric

Similarly to [HHL+23] and [HTDX22], we use LPIPS (Learned Perceptual Image Patch Similarity) [ZIE18a] to evaluate the diversity of the images generated by Stable Diffusion.

We measure the pairwise similarity between images in a batch and compare the obtained values for different modifications of Stable Diffusion generation algorithm.

3.3.3 Ethnicity and gender classification for images portraying humans

As mentioned in [The23], Stable Diffusion may lack diversity in ethnicity representation. That is why, for the prompts that we use for the human face generation, we compare ethnic diversity in between our methods and basic version of Stable Diffusion. In order to identify the ethnicity of a person present on an image, we use DeepFace ethnicity recognition [TYRW14]. In particular, we identify the following groups of ethnicities:

- Black,
- Asian,
- Hispanic,
- White or Middle Eastern.

For gender classification (male/female), we also use DeepFace.We compute the percentage of batches where all pairs of genders and ethnicities are present or at least 3 out of 4 ethnicities are present (similarly to colors).

3.3.4 Multiplicative improvement

Our method aims to increase representation of underrepresented groups. That is why for all the metrics mentioned above we compute percentage versus multiplicative improvement score. Here percentage stands for the percentage of batches that follow some characteristic C (for example, contain images of Asian men) and multiplicative improvement stands for multiplicative increase in this percentage for our preferred method (**"pooling_cap"**) compared to the baseline method (**standard Stable Diffusion**). By using this metric, we can evaluate our efforts in promoting the inclusivity of underrepresented classes.

3.4 Experiment setting

As a baseline, we use Stable Diffusion v-5 with PNDM Scheduler.

We measure diversity using LPIPS or artificial classes (image hue) or human-centered criteria (gender/ethnicity) and check various batch sizes. We use machines with 8 Tesla V100-SXM2-32GB GPUs and 80 x86_64 CPUs.
The full code is provided in https://anonymous.4open.science/r/DiverseDiffusion-1012.

3.4.1 Experiments on small batches

For our small batch experiment setup, we choose the following list of prompts:

- "face", '
- 'rose",
- "butterfly",
- "cat",
- "horse",
- "car",
- "ornament",
- "bird",
- "color",
- "a professional photograph of an adult person face",

- “photo of an animal in the grass” and
- “octane, hyperrealistic, backlit”.

In each experiment, we consider batches of 3, 5 and 10 images in order to compute diversity metrics in each of these batches, and to compare our modifications with original Stable Diffusion. For each batch size and each modification, we create at least 2500 batches.

3.4.2 Experiments on big batches

For our big batch experiment setup, we choose the following list of prompts:

- “a professional photograph of a man face”,
- “a photograph of a person with different colored eyes”,
- “a passport-style photograph of a person’s face”,
- “a professional photograph of an adult person face”,
- “a close-up photograph of an elderly person’s face” and
- “a beauty shot of a model’s face”.

These prompts are all centered on human photos and thus allow us to evaluate not only LPIPS and color diversity but also gender and ethnicity variation, which are crucial to ensure in any human-centered applications of diffusion models. In each experiment, we consider batches of 50 images in order to compute diversity metrics in each of these batches and compare our modifications with the baseline: original Stable Diffusion. For each modification, we create at least 900 batches.

3.5 Experiments and results

An example of image batches generated with our method “pooling_cap” and original Stable Diffusion is presented in Figure 3.2. In two randomly chosen examples among all batches generated by the baseline and “pooling_cap” for the prompt “a photograph of a person with different colored eyes” and batch size 50, we can see that while getting images corresponding to the prompt remains difficult for our chosen method, we still get improvement both in number of images corresponding to the prompt and ethnic diversity.

Further, we present the experimental evaluation of our proposed methods for promoting diversity in generated image batches. We assess the color diversity, gender and ethnicity representation, and image diversity using

Figure 3.2: Examples of images generated with the prompt "a photograph of a person with different colored eyes": pooling_cap against baseline Stable Diffusion, batch size=50. Images corresponding to the "expected output" of the prompt are highlighted in red.

the LPIPS metric. The experiments are conducted on various prompts and compared against the baseline Stable Diffusion method.

3.5.1 Color evaluation

In this subsection, we evaluate the color diversity of the generated image batches. We present multiplicative improvement results, summarizing the color diversity improvement for various values of the coefficient

K across different prompts specified in Section "Experiments on small batches". Other experiment results are provided in the Appendix B.

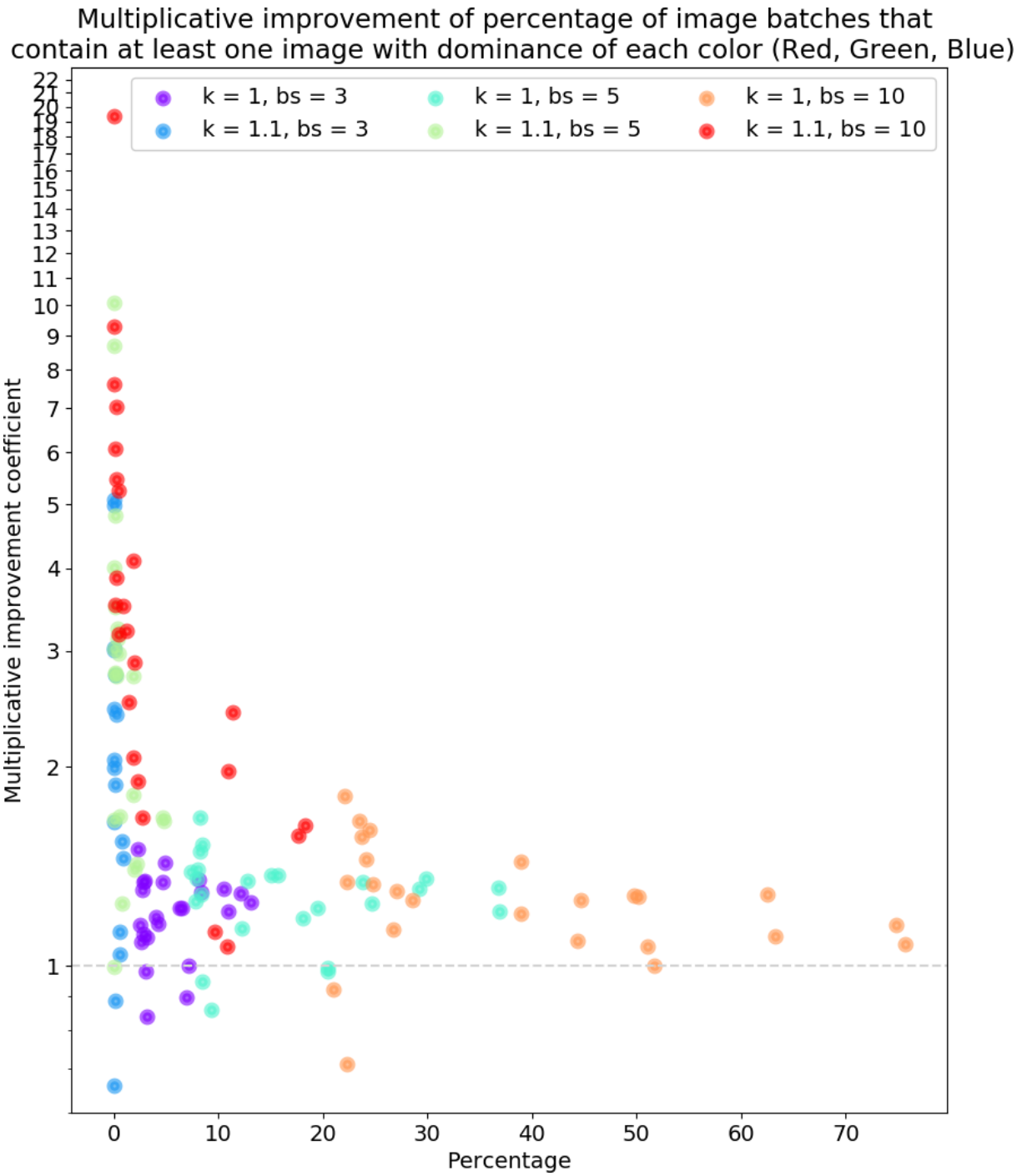

Figure 3.3: Multiplicative improvement of percentage of batches containing images with all 3 dominant colors: pooling_cap against baseline Stable Diffusion, depending on K as specified in the text and batch size bs. Improvements are greater on the left, *i.e.* more difficult cases.

In Figure 3.3, we can see that in majority of the cases using pooling_cap method does not decrease the representation of batches with all 3 colors dominance. We also can see that we get very high (> 2.5) improvement coefficient numbers in cases where color dominance is defined using coefficient k = 1.1, which significantly increases the number of batches featuring this rare characteristic.

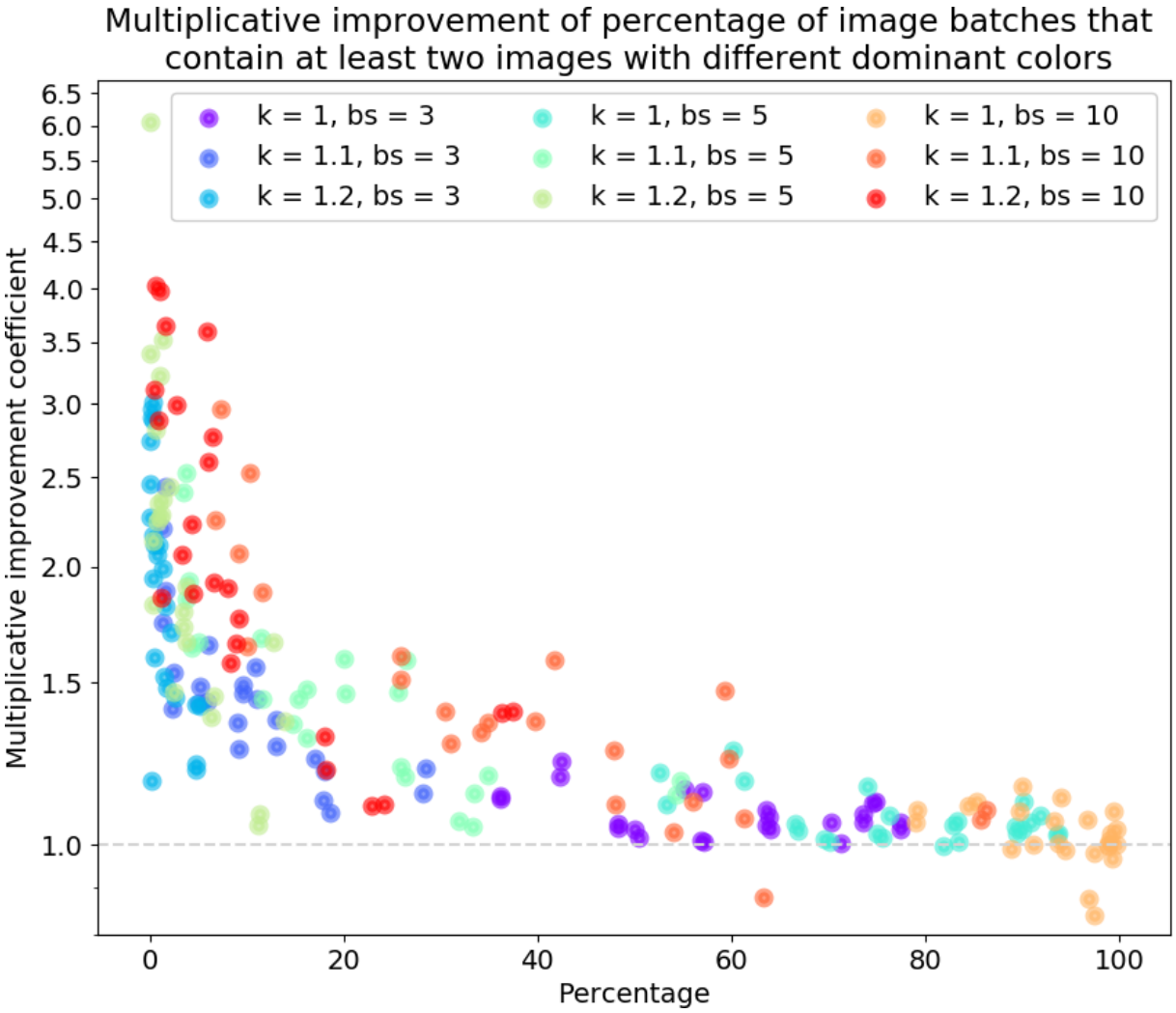

Figure 3.4: Multiplicative improvement for the percentage of batches containing images of at least 2 of the 3 categories: the improvement is greater for more difficult cases, which are on the left (pooling_cap against baseline Stable Diffusion method).

In Figure 3.4, we can see that in majority of the cases using pooling_cap method does not decrease the representation of batches featuring at least 2 dominant colors. Similar to figure 3.3, we observe very high (> 2.5) improvement coefficient numbers only in cases where color dominance is

defined using coefficient $K > 1$, which significantly increases the number of batches featuring this rare characteristic. We also can see that for modes where having at least 2 dominant colors per batch is a rare characteristic (less than 50% of batches feature it), we always have improvement coefficient > 1, thus increasing the color diversity.

3.5.2 Gender and ethnicity evaluation

Here, we evaluate the impact of our proposed method on the diversity of gender and ethnicity representation in the generated image batches.

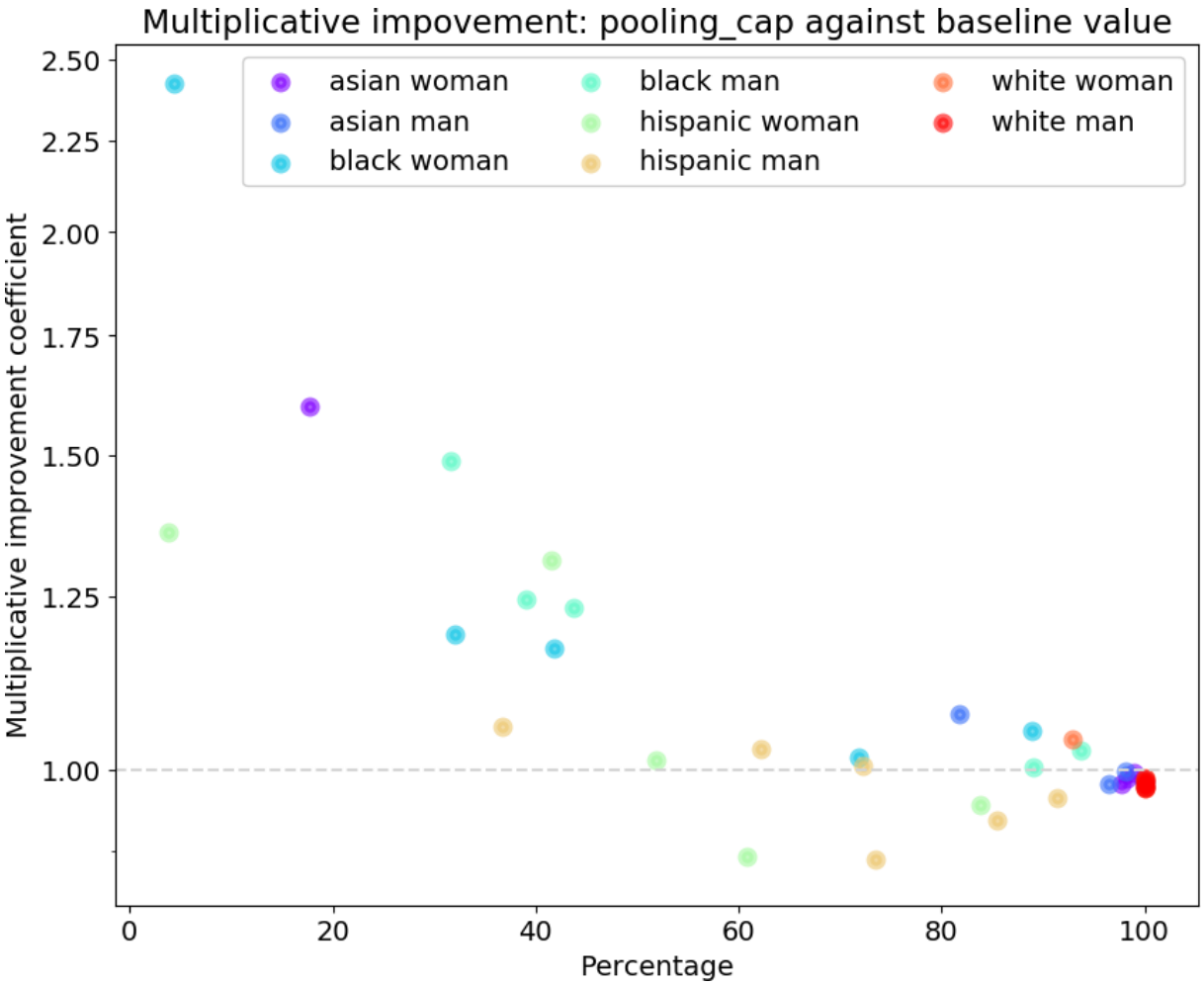

Figure 3.5: Multiplicative improvement (increase) of percentage of batches containing various ethnicity + gender pairs: we get a greater improvement for more difficult cases (pooling_cap against baseline Stable Diffusion method).

Our results, as illustrated in Figure 3.5, demonstrate a remarkable enhancement in the representation of underrepresented categories. This im-

provement is consistently observed across all ethnicity/gender pairs that initially appeared in less than 60% of the batches. Notably, our cap pooling technique has led to a substantial increase in their presence, with certain categories being up to 2.4 times more prevalent than in the standard Stable Diffusion version.

While we see some detrimental results for Hispanics representation, it is important to note that ethnicity identification based solely on a person's image is not always accurate for artificial intelligence methods. In this research, we primarily focus on the diversity between white and black individuals, as they can be visually distinguished with minimal ambiguity [MTB+22]. In Figure 3.5, we observe a positive improvement in the representation of black individuals, both male and female, compared to the baseline results for Stable Diffusion.

Overall, these findings highlight the effectiveness of our approach in promoting diversity and addressing underrepresentation in gender and ethnicity across various image batches.

3.5.3 LPIPS evaluation

In this subsection, we evaluate the generated image diversity using the LPIPS metric. LPIPS measures the perceptual similarity between images based on deep neural network representations. A lower pairwise LPIPS score indicates greater diversity among the generated images.

Here we compare the performance of different methods, including the baseline and our proposed "pooling_cap" method, across different batch sizes.

Figure 3.6 presents the average batch pairwise LPIPS distance for a batch size of 3. We can see that for most of the experiments "pooling_cap" method proves to be more diverse than the baseline, while for certain cases such as "face", baseline outperforms "pooling_cap" . We also notice that "pooling_cap" is not the single method that performs well. Others, such as "cap", prove to be the best for various prompts as well. For example: "rose", "ornament" and "car".

Figure 3.7 illustrates the average batch pairwise LPIPS distance for a batch size of 5. We can see that for most of the experiments "pooling_max" method proves to be most diverse among all the methods outperforming "pooling_cap", while "pooling_cap" is still better than the baseline for most of the cases. Similarly to Figure 3.6, we can also notice that for the cases where baseline had a small average LPIPS distance (for instance "octane" and "photo of an animal"), all the proposed methods significantly outper-

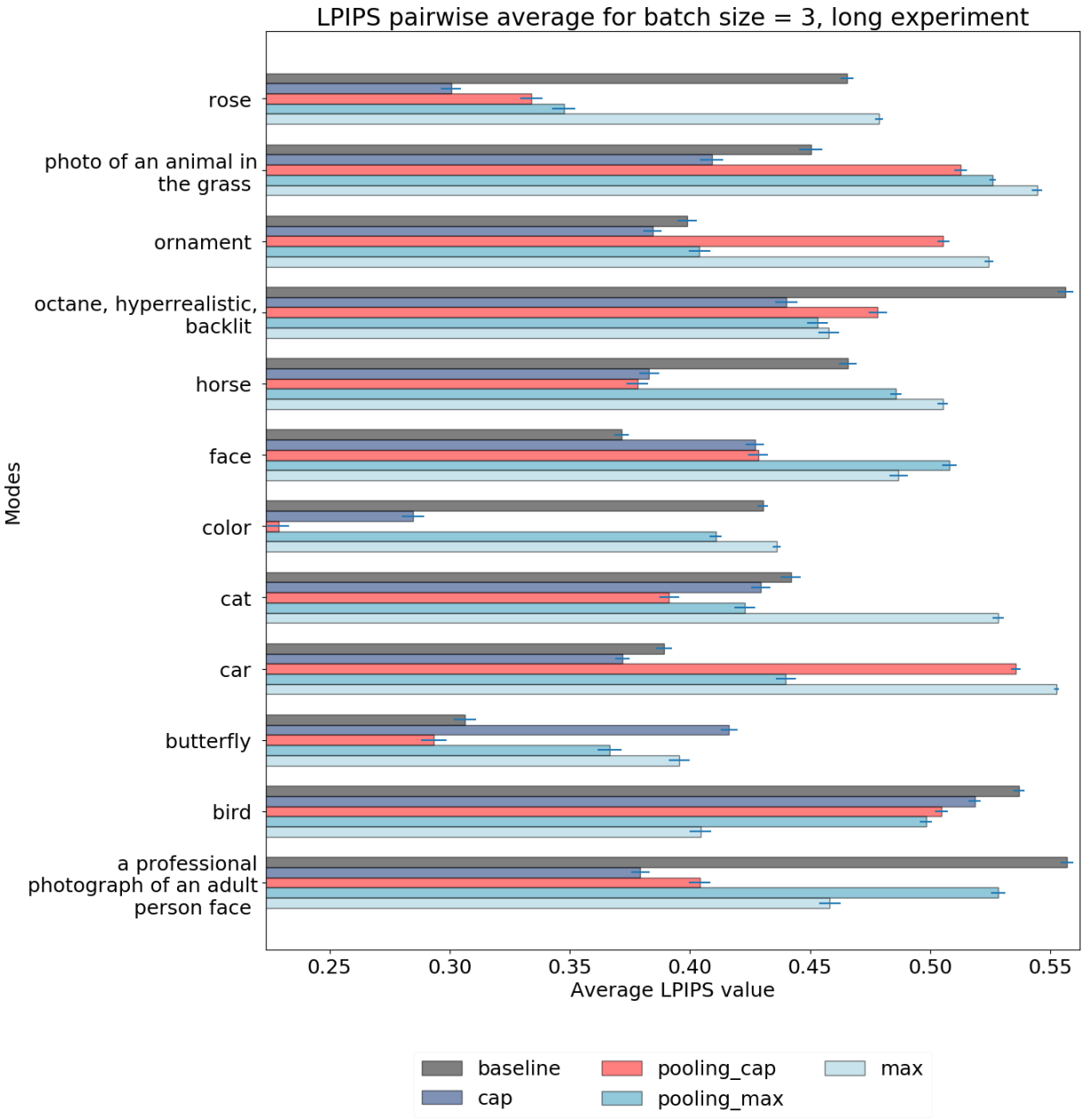

Figure 3.6: Average batch pairwise LPIPS, batch size=3 : no clear conclusion overall, due to the small batch size.

form the baseline. This fact shows that even for small batches, our methods provide diversity benefits in cases when it was really lacking.

Figure 3.8 showcases the average batch pairwise LPIPS distance for a batch size of 10. Here, we can see that for most of the experiments "pooling_max" is the most diverse among all the methods, while "pooling_cap" is still better than the baseline for most of the cases. We again notice that in cases where the baseline had a small average LPIPS distance (for

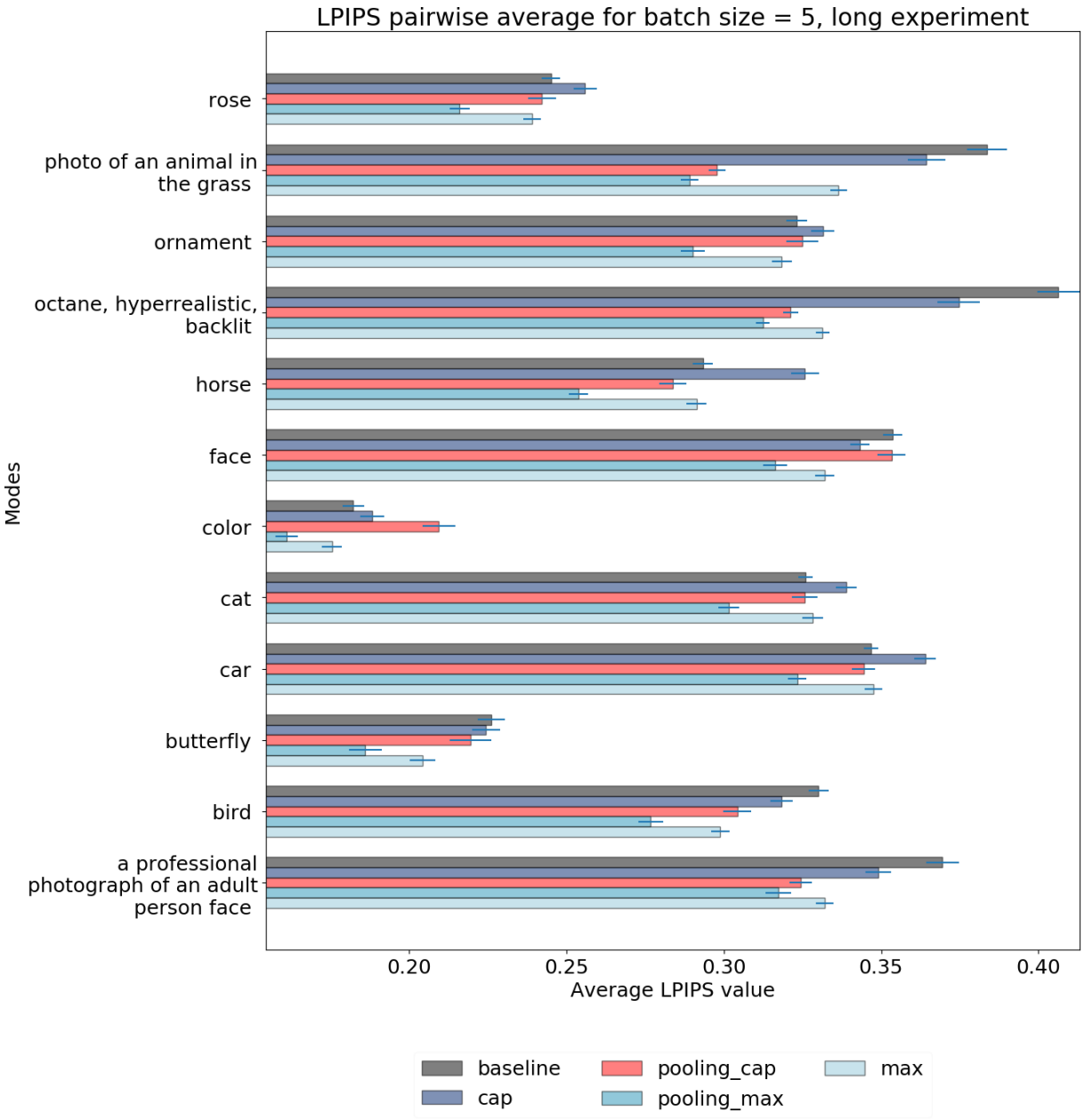

Figure 3.7: Average batch pairwise LPIPS, batch size=5: the two Pooling methods are often better than the baseline, though results are clearer for batch size 50.

instance "octane"), all of the proposed methods significantly outperform the baseline. This fact shows that even for small batches, our methods provide diversity benefits in cases when it was really lacking.

In contrast to the results presented in Figure 3.6, in Figure 3.8 "pooling_max" method shows to always outperform the baseline. This observation indicates that our methods provide better diversity guaranties for the

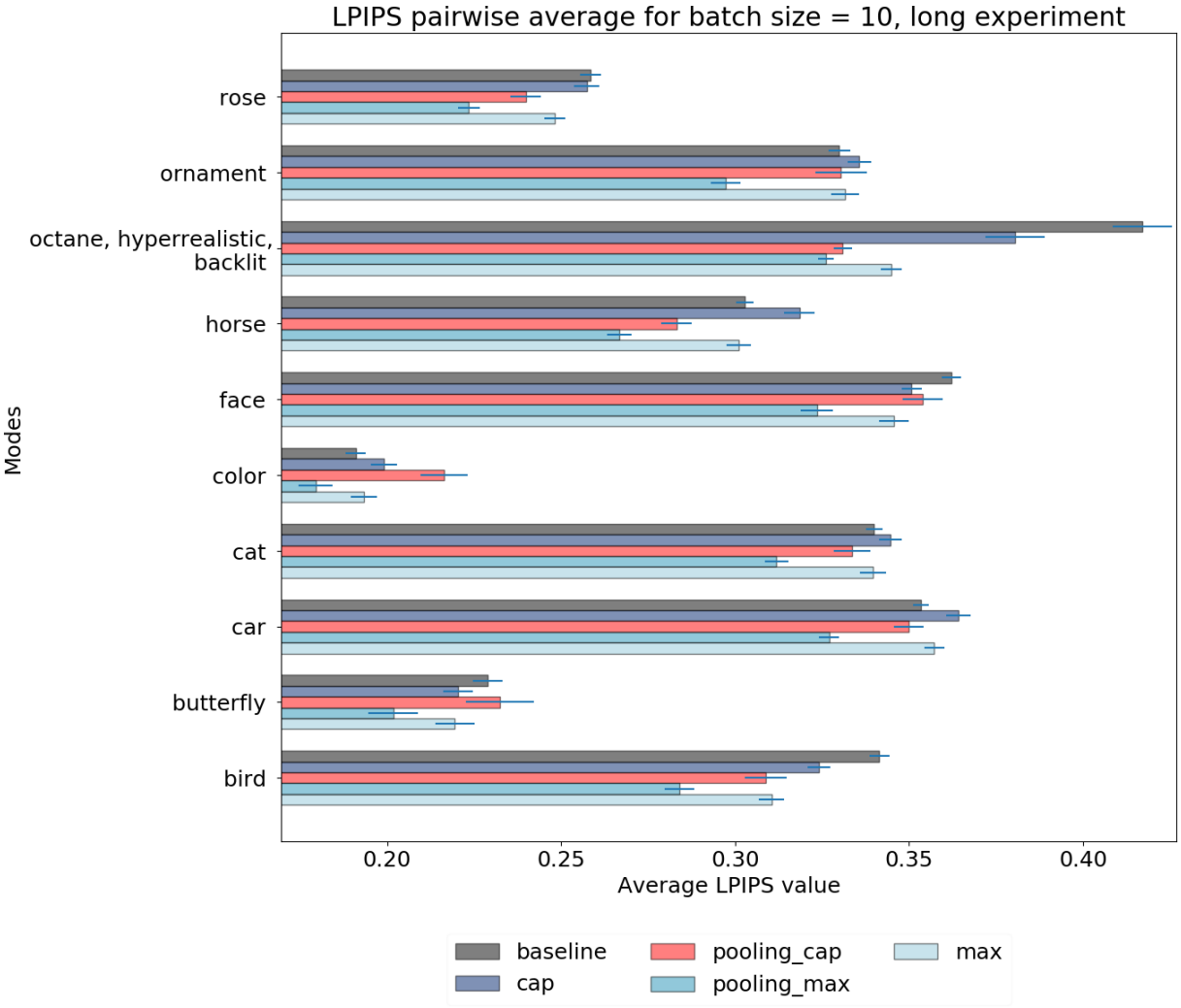

Figure 3.8: Average batch pairwise LPIPS, batch size=10. The two Pooling methods are often better than the baseline, though results are clearer for batch size 50.

bigger batch sizes.

In contrast to the results shown in Figures 3.6, 3.7 and 3.8, in Figure 3.9 we can see that the "pooling_cap" method is consistently highlighted as the most diverse across all experiments. This emphasizes that larger batch sizes lead to better diversity due to the increased availability of reference points for each new image. This experiment strengthens our earlier observations on color diversity and solidifies the superiority of the "pooling_cap" variant over the standard Stable Diffusion approach in terms of diversity.

For the experiment presented in Table 3.1, we compute LPIPS distance between several image pairs for the same prompt (not necessarily in the same batch) and then average it between different prompts.

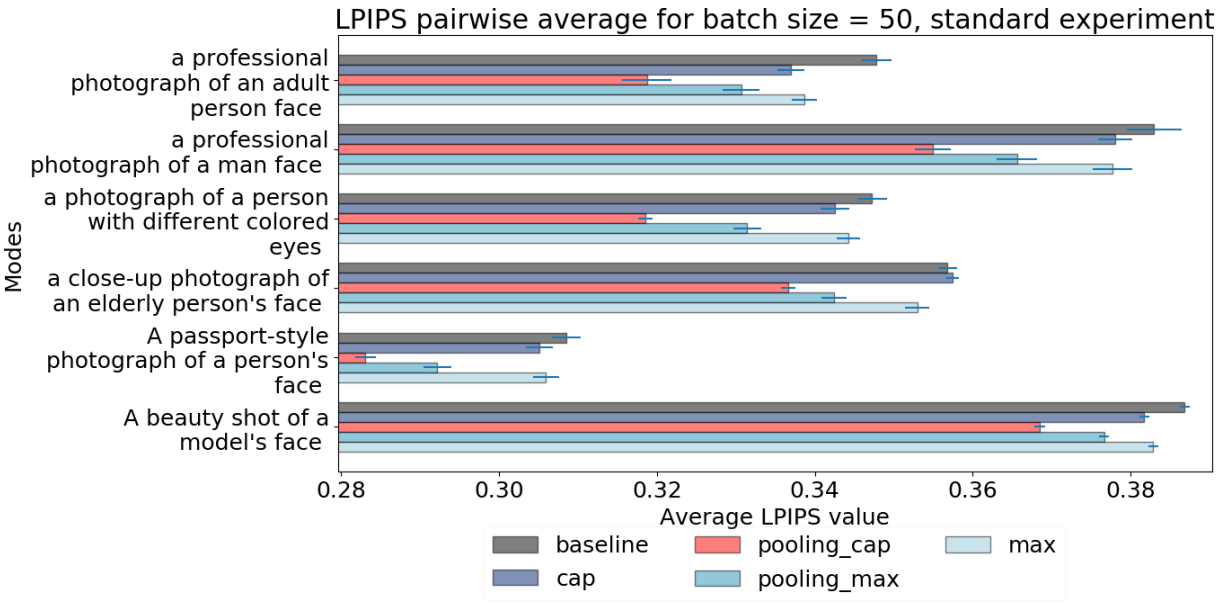

Figure 3.9: Average batch pairwise LPIPS, batch size=50. For this batch size,in all cases pooling methods (both max and cap) are beneficial to diversity as measured by LPIPS.

Method	average LPIPS
Baseline	0.354 ± 0.004
ENTIGEN	0.325 ± 0.004
Pooling_cap	$\mathbf{0.294 \pm 0.005}$

Table 3.1: Average pairwise LPIPS value across different batches with the same prompt.

We do this experiment across standard Stable Diffusion, our preferred method "pooling_cap" and another diversity method and ENTIGEN [BYMC22] diversity method applied to the same prompts as in subsection "Experiments on big batches". More precisely, to each prompt, we add either "irrespective of their gender" or "irrespective of their color". We can see that even though our generation is focused mainly on batch setting, our method achieves better diversity both than Stable Diffusion and ENTIGEN, that was specifically designed to eliminate human-centered biases such as gender and ethnicity.

3.6 Conclusions

In conclusion, our contributions in this chapter include

1. a general technique that can be applied to existing text-to-image models to increase image diversity, which works in an unsupervised manner

with a negligible overhead and

2. experiments that showcase the diversity advantages of our proposed approach applied to Stable Diffusion, through classification (both image hue and gender/ethnicity) and LPIPS measurements.

Experimental results highlight the impact of our proposed method on color diversity. By analyzing the multiplicative improvement in the batches containing images with dominant colors, we observe that the "pooling_cap" method consistently maintains or increases the representation of batches with all three dominant colors. Additionally, we notice significant enhancement when using the color dominance coefficient $K > 1$ (*i.e.*, cases in which success is rarer, hence the problem is more difficult). These findings validate the effectiveness of our approach in promoting diverse color compositions in generated images.

Our experimental results consistently demonstrate the superiority of our "pooling_max" and "pooling_cap" methods over the baseline (standard Stable Diffusion) in terms of average pairwise LPIPS distance within batches. This evaluation metric serves as a reliable measure of image diversity, and our methods consistently outperform the baseline, showcasing their ability to generate more diverse images. These findings highlight the effectiveness of our approach in expanding the range of image variations and improving overall diversity.

Furthermore, the results show a notable improvement in the representation of underrepresented categories across different ethnicity/gender pairs. The "pooling_cap" method led to a substantial increase in the presence of these categories, with some categories being up to 2.4 times more frequent, compared to the baseline. It is important to acknowledge the inherent limitations of ethnicity identification based solely on visual cues, but our focus on improving the representation of underrepresented categories, particularly between white and black individuals, aligns with the goal of promoting diversity and inclusion in image generation.

4

Generative Image Privacy

4.1 Privacy in the age of expansive data

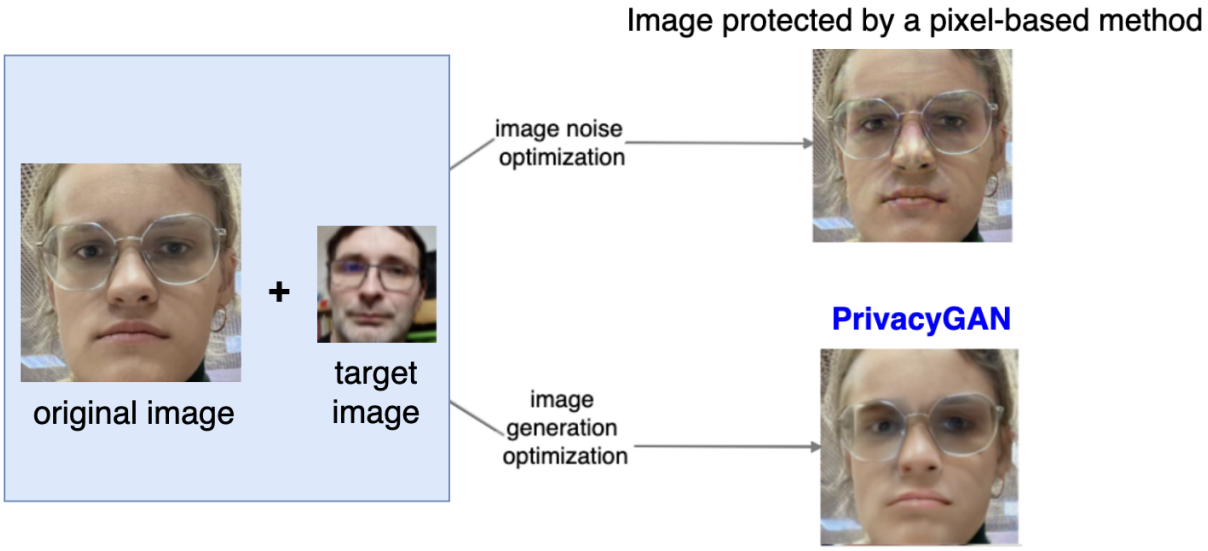

Figure 4.1: Schema for both data poisoning and generative privacy methods. We take the original image (OI) and create an image that is recognisable by human observers while being unlikely to be identified by image recognition methods using
(i) classic pixel-based approach:
adding pixel noise such that it makes the modified image in the embedding space closer to the target image than to the original image,
(ii) (our approach, PrivacyGAN):
generation of visually similar but distant images in the embedding space.

In this chapter, we propose a general approach to the use of generative methods for privacy. We train our methods to be effective for embedding methods, on which pixel-based methods such as Fawkes used to fail. Our goal is to create images that resemble original photographs and are suitable for sharing on social media platforms, while also preventing identification by modern image recognition neural networks without using anonymization. To achieve this, we explore the effectiveness of two generative methods: a generative adversarial neural network StyleGAN [KLA19b], and the autoencoder VQGAN [ERO21]. These generative methods are known for their realistic image generation capabilities, which adds an added layer of difficulty for neural network recognition.

By modifying facial images using generative methods, we aim to preserve their recognizability to human observers while rendering them unrecognisable to many existing image recognition neural networks. Inspired by pixel-based methods like Fawkes [SWZ+20a], we propose modifying the generated "private" images towards a different target image in the embedding space and evaluate the robustness of our approach against unknown image recognition neural networks.

We validate our privacy methods on the Labeled Faces in the Wild (LFW) dataset [HMBLM08], as well as introduce a new dataset of face crops extracted from Casual Conversations [HBD+21] to ensure their effectiveness in various environments. In Fig. 4.1 we present the schema of image modification using both pixel-based and generative methods.

Our proposed generative tools make subtle modifications to user images without adding pixel noise, so the resulting photos look natural and protect user privacy. Our algorithms operate within a black-box framework and demonstrate their efficacy against image recognition techniques they were not specifically trained on. We offer flexibility in selecting methods and privacy settings and conduct a comparison between our approach and existing state-of-the-art privacy protection methods, such as Fawkes.

To sum up the contributions of this chapter, we

- propose a novel approach to facial image privacy based on generative methods;
- create a new privacy evaluation approach based on the percentage of dataset images that are closer in an embedding space to a modified "private" image than to an original image;
- propose a new facial image dataset extracted from the Casual Con-

versations dataset [HBD+21] videos;

- evaluate the privacy of the modified images against various embedding methods (including transfer to embeddings not used in our privacy method) and provide human evaluation of image quality for state-of-the-art and novel privacy methods.

4.2 Privacy Algorithms

4.2.1 A well-known pixel-based method: Fawkes.

Fawkes [SWZ+20a] is a data poisoning method that presents subtle image perturbations to the images. One of its main features is the concept of target image: by suggesting elements from a side target image, Fawkes ensures that the modified image will be recognised as another person by neural networks, thus ensuring privacy. Unlike just maximising the distance between the embedding of the original image and the modified image, this method

- helps to keep the embedding of modified images within a valid range for a given dataset and
- ensures that the embedding of the modified image does not stay close to its original version in the given dataset.

The idea of Fawkes is to pair each original image (**OI**) with a target image (**TI**). Then Fawkes associates to each original image a 'cloak image' (**CI**) which consists of noise obtained by optimising the following loss:

$$L_{Fawkes} = ||emb(TI) - emb(OI \oplus CI)||,$$

where:

- **OI** is an original image;
- $\oplus$ is capped addition;
- *emb* is an embedding method used for cloak optimisation in order to obtain a modified "private" version of an original image;
- **TI** is a target image; the private version of OI should be labelled the same as TI by the chosen image recognition system;
- ρ is a parameter that caps the noise strength;
- **CI** $< \rho$ is a cloak or a noise that should be added to OI in order for it ensure its' privacy;

- $OI \oplus CI$ is the published rendition of OI.

In our experiments, by default, we use the "high" mode of Fawkes (as mentioned in [SWZ+20b] and [SWZ+20a]), since it provides a decent level of protection, and it is possible to compare this setting of Fawkes with our methods.

4.3 Privacy algorithms: generative makeup transfer method AMT-GAN

In this chapter, we compare the results of our method PrivacyGAN based on generative techniques such as StyleGAN [KLA19b] and VQGAN[ERO21] to a method for generative makeup transfer known as AMT-GAN [HLZ+22].

The objective of AMT-GAN is to produce adversarial images that incorporate the makeup style of reference images. Although AMT-GAN introduces more alterations to the original image, it confines these modifications to the makeup application areas, thus resulting in visually natural images, as demonstrated by the FID results. The authors of the method employ LPIPS [ZIE+18b] loss to retain image similarity to the original, which we also utilise in our work.

In the main part of the chapter, we mention that while photographs of individuals wearing makeup may look appropriate and natural to some people, there are certain drawbacks to this approach. Firstly, publishing photos with makeup may be deemed unsuitable for certain groups of individuals. Secondly, if the face in the photograph is not clearly discernible, AMT-GAN may not recognise it and may not generate a private version of the image, unlike Fawkes [SWZ+20a] and our method.

4.3.1 Our proposed generative methods based on VQGAN and StyleGAN

Our proposal involves utilising generative models for privacy protection, with a focus on generating an image that closely resembles the original in visual appearance while safeguarding users against image recognition attacks. Our objective is not to anonymize the image. In this work, we expand on the idea of target images introduced in Fawkes. We use target images to ensure that the modified version of an image is closer to the target image than the original image in the chosen embedding space.

To select target images, we choose from images in the dataset that have not been used in experiments. For each specific image, we select a target image based on its distance from the original image in the chosen

embedding method used for optimization. The chosen target image should be far enough from the original image to ensure effective privacy protection.

We select the loss function L based on the goal of preserving the identity of the original image (OI) for humans while ensuring that the generated image (GI) embedding is as close as possible to the distant target image embedding. For this purpose, we use the Learned Perceptual Image Patch Similarity (LPIPS) distance for the preservation of OI identity and optimise the embedding distance between the generated image (GI) and the target image (TI) to achieve the closest possible embedding.

The loss for any generated image always consists of the sum of the following parts:

- LPIPS distance between generated and original image
- For each of the embeddings used for optimisation, coefficient K multiplied by the mean squared distance between the modified "private" image embedding and target image embedding.

$$L_{\substack{generative \\ privacy}} = LPIPS(OI, GI) + K \times \sum_{\substack{emb \in \\ embeddings}} || \, emb(GI) - emb(TI)||.$$

The hyperparameters of the loss described are

- the coefficient K (*i.e.*, weight compared to LPIPS) for the embedding distance,
- the learning rate,
- the batch size, and
- the number of iterations.

4.3.2 Generative Privacy Algorithm: PrivacyGAN

Here, we describe PrivacyGAN, an algorithm that we propose for creating private versions of facial images.

As input to the algorithm, we use an original image **OI** and a target image **TI**. The algorithm aims to produce an image **GI** which would be a "private" version of OI, unrecognisable by many image recognition neural networks. In order to do that, we are using generative methods such as StyleGAN and VQGAN. TI is chosen randomly among the images furthest in embedding space from OI.

Algorithm 5: PrivacyGAN: Private image generation

Require: OI: Original Image
Ensure: GI: Generated Image
$TI \leftarrow$ Target Image {(distant from OI in the embedding space)}
$z \leftarrow$ random
$G \leftarrow$ image generation method
$K \leftarrow$ optimisation coefficient
chosen_emb $\leftarrow$ list of embedding methods for optimisation (we distance GI from OI in these embedding spaces)
for i in range(0, num_iterations) **do**
  $GI \leftarrow G(z)$
  emb_dist $\leftarrow 0$
  for *emb* in *chosen_emb* **do**
    emb_dist $+= ||emb(GI) - emb(TI)||$
  end for
  lpips_dist $\leftarrow LPIPS(GI, OI)$
  $loss \leftarrow lpips\_dist + K \cdot emb\_dist$
  $z \leftarrow update(z, loss, \nabla_{\text{loss}})$
end for
return GI

The algorithm consists of an iterative optimisation process, where **num_iterations** represents the number of iterations and **G** is the image generation method. In the latent space of G, we find a latent variable **z** and generate an image G(z), which we refer to as GI. For each of the embedding methods in the set **chosen_emb**, in each iteration of the algorithm, we compute the embedding distance between GI and TI, as well as the LPIPS distance [ZIE+18b] between images GI and OI. We use the computed distances to calculate the $L_{\substack{generative \\ privacy}}$ that we mention as 'loss' in the algorithm.

4.4 Evaluation Methods

4.4.1 Metrics for Privacy

In order to evaluate the privacy of generated images, it is important to determine how far the generated image is from the original in the dataset. After applying an image recognition neural network, attackers may choose to verify if the person on the image matches the top few possible results.

That is why the method would work better for privacy protection if:

1. the modified image would not be recognised as its original version; and
2. the original image would be far away from the modified one in the embedding space.

We ensure 2. not only by using existing evaluation methods such as Recall@k that help us to make sure that the modified "private" image is far from the original in absolute values but also by introducing a novel evaluation method that verifies the original and modified image being far in embedding space relative to the dataset size.

To measure the distance from the original image to its modified private version, we use the following privacy metrics: Recall@k and $Percentage$.

Recall@k for the set of query images L (which can either be original or modified images) and test images M, is defined as

$$Recall(L, M, k) = 100 \frac{\sum\limits_{q \in L} 1_{Id(q) \in Id(N(q,k,M))}}{||L||},$$

where

- function Id maps the set of people's images to the set of (unique) identities of the individuals present on these images;
- function $N(q, k, M)$ returns a set of k images from M that have the closest embedding to the one of the query image q.

We propose the use of a new metric, called the "Percentage", in addition to the Recall metric, to evaluate the effectiveness of our privacy methods. The reason for introducing this new metric is to ensure that the modified image is not only far from the original image in absolute terms, as ensured by Recall@k, but also in terms of the percentage of dataset size. This provides a common privacy metric that can be used to compare the effectiveness of our methods across different dataset sizes.

$Percentage$ is the proportion of images for each query image from the dataset L in between the query image and the closest image with the same identity from the dataset M:

$$Percentage(L, M, k) = 100 \sum_{q \in L} \frac{Between(q, N(q, 1, M))}{||L|| \times ||M||},$$

where

- function $Between(q_1, q_2)$ returns the number of images in the dataset M that have a smaller distance to the embedding of q_1 than the distance in-between the embeddings of q_1 and q_2.

4.4.2 The problem of transfer

In practical scenarios, it is crucial that privacy methods are effective against various image recognition neural networks. We optimise our privacy methods to be effective for specific embeddings, and transferring to a different embedding method can be challenging as new methods are continually emerging. It is impossible to guarantee that privacy methods will be effective against future attacks, as some methods have been broken by newer recognition neural networks [RDT21]. To evaluate the effectiveness of our proposed methods, we conducted two sets of optimisation experiments.

The first experiment involves optimising StyleGAN and VQGAN image generation to be effective against the FaceNet embedding method. We aim to make the FaceNet embedding of the generated image distant from that of the original image during the optimisation process. We compare our proposed methods to Fawkes, which uses the same embedding method for optimisation, in Table 4.1. Additionally, we aim to test the transferability of our methods to embedding methods introduced after FaceNet, which Fawkes does not prove to be effective against [RDT21].

The second experiment involves optimising StyleGAN and VQGAN image generation using MagFace and MobileFaceNet embedding methods, which have been shown to increase the robustness of generated images. We also compare them to the makeup transfer method AMT-GAN [HLZ+22]

4.5 Datasets and embeddings

4.5.1 The Labelled Faces in the Wild

The dataset [HMBLM08] contains multiple images for each person, with the number of images per person varying between 1 and 530. To ensure fairness, we extract a sub-dataset from the original dataset, which includes 5 randomly chosen images per person. We exclude images of people who have less than 5 photos present in the dataset. This sub-dataset is referred to as LFW in the following sections of this work.

4.5.2 The Casual Conversations dataset

The Casual Conversations dataset (CC) [HBD+21] comprises 45186 videos, each of which features one person.

We select 997 videos and extract 5 face crops of size 456×456 per person present in the dataset (in case the video contains face crops of a required size).

The process of selecting these face crops is as follows:

- We select all the time frames from the video featuring a specific person.
- We check if, among these time frames, there are at least 5 non-consecutive (± 10) time frames that satisfy the following conditions:
  - they contain a face crop of a size at least 456×456 with a margin of size 100;
  - average brightness of a time frame is at least 70. This condition is required since, among the videos in the CC dataset, there are many that were recorded in complete darkness, and it is not realistic to have such face crops as profile pictures.
- If there are more than 5 time frames selected, we randomly choose 5 of them and add them to the dataset.

We make sure that we don't select successive frames of the video since they could contain identical face crops. For the confounders set, we randomly choose different people's face crops that also satisfy conditions 1 and 2 and do not feature a person who was already selected for our primary dataset before.

The key difference between our novel dataset and LFW is that faces in our proposed dataset have similar backgrounds and are taken within a short timeframe, creating an additional challenge for privacy protection. That lack of variety makes this particular dataset very interesting for our research. By testing our methods on it, we are able to ensure that, even if there are many very similar photos of the same person in the dataset, the proposed privacy tools can still be effective. It is particularly important in cases where people publish their images from similar locations on different platforms over the internet.

4.5.3 Our proposed methods for transfer to unknown embeddings: optimising on multiple embeddings

The embedding methods that we are using in this work are the following:

- FaceNet [SKP15];
- ArcFace [DGXZ19];

- SphereFace [LWY+17];
- MagFace [MZHZ21];
- MobileFaceNet [CLGH18] with implementation from the FaceX-Zoo library [WLH+21] and
- ResNet_152 [HZRS16] with implementation from the FaceX-Zoo library [WLH+21].

We evaluate the effectiveness of our proposed privacy methods in a black-box setting (*i.e.*, robustness to unknown image recognition methods not used in the privacy method). We optimise the generated image for one or two embeddings from the list and then check the generated image against all the other embeddings. Thus, we make sure that our privacy methods transfer well to unknown embeddings and can be used for the privacy protection of real photos published online.

4.6 Experiments and Results

Here we define the settings and notations that we use in our experiments.

We set the hyperparameters of the generative privacy loss ($L_{\substack{generative \\ privacy}}$) for our experiments in the following way: the learning rate to 0.01 and the batch size to 32. The only parameters that we modify from experiment to experiment are the coefficient K and the number of iterations.

We have introduced the notations "o.i" and "m.i" to represent the original image and the modified image generated by any privacy-preserving algorithm, respectively. For our recall evaluation, we select either

- the original image **(o.i. context)** or
- the modified image **(m.i. context)**

as the query image, where the dataset used for recognition includes modified images and confounders for the former and includes the original images and confounders for the latter.

In all metric calculations, we also use a set of confounders, which are not used as queries in our experiments and are sourced from the same dataset as the original images. The number of confounders is always less than or equal to $\frac{1}{5}$ of the number of original images. To compare privacy protection methods, we use the average value of the transfer recall (*i.e.*, Recall@10) for all embeddings for which the algorithm was not optimized. Incorporating

confounders in our experiments brings us closer to real-world scenarios, where datasets may contain unrelated images that can potentially affect the experiment results.

Moreover, in order to compare VQGAN, StyleGAN, and Fawkes to one another, we use different sets of parameters chosen to match the transfer recall results. Thus, we are able to compare the generated image quality and see which of the methods generates the best images in terms of image quality for a given privacy performance (measured by recall).

Later in this section, we use the following notations:

- By **standard version of StyleGAN**

  we mean PrivacyGAN equipped with StyleGAN optimized with a coefficient $K = 0.03$ for embedding distance in the loss and 128 iterations;

- By **standard version of VQGAN**

  we mean PrivacyGAN equipped with VQGAN optimized a coefficient $K = 0.03$ for embedding distance in the loss and 1000 iterations;

- By StyleGAN_x_y / VQGAN_x_y

  we mean PrivacyGAN equipped with StyleGAN/VQGAN optimized a coefficient $K = x$ for embedding distance in the loss and y iterations.

4.6.1 Experiment 1: Comparing Pixel-Based and Generative Methods Optimised for One Embedding on the LFW Dataset

In Table 4.1, we compare standard versions of VQGAN and StyleGAN and their versions StyleGAN_0.003_500 and VQGAN_0.005_128 that we prepare specifically to match the privacy results of Fawkes. With this parametrization, they have the same transfer recall score, which allows us to compare fairly the image quality of our proposed generative methods and the state-of-the-art pixel-based method Fawkes.

The transfer recall values (average values of Recall@10 for other methods from the list) are

- 62.16% for StyleGAN,
- 89.28% for StyleGAN_0.003_500,
- 65.49% for VQGAN,
- 90.07% for VQGAN_0.005_128,

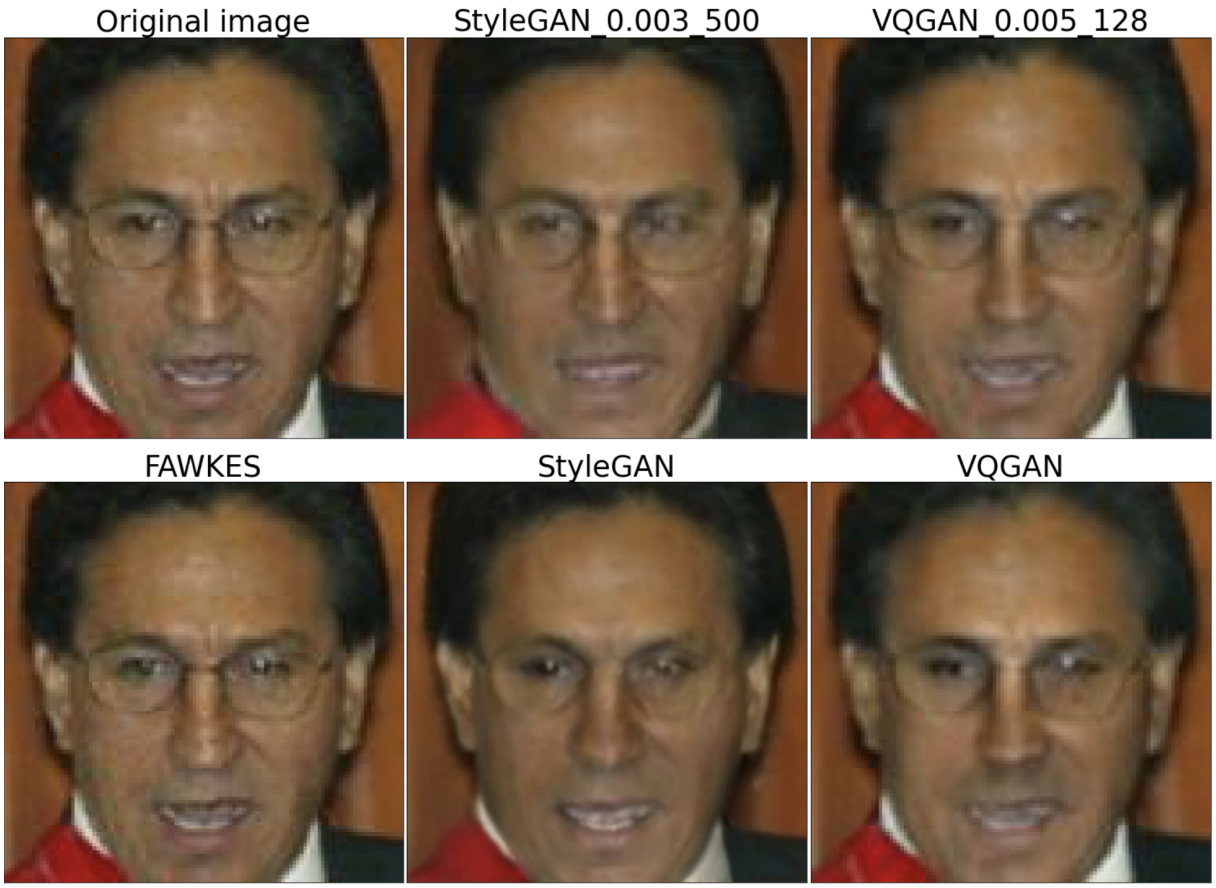

Figure 4.2: Experiment 1: Examples of original images from the LFW dataset and their counterparts modified by different privacy methods: StyleGAN_0.003_500, VQGAN_0.005_128, StyleGAN, VQGAN (using FaceNet as an embedding method for optimization), and Fawkes. Here we can see that, while generative methods in general add more modification to an image than Fawkes, generative methods produce realistic images and do not add pixel noise. Study based on human ratings in Table 4.8.

- 90.90% for Fawkes.

In order to make sure that image privacy is robust against various facial recognition systems, we study a transfer to different embedding methods (ArcFace, MagFace, SphereFace, MobileFaceNet, ResNet_152). Some results are in Table 4.2 (SphereFace) and in Table 4.3 (MagFace).

More results can be found in the Appendix C.

Examples of original images from the LFW dataset and their modifications obtained by our methods and by Fawkes are presented in Fig. 4.2. More examples are in the Appendix C.

Overall, from Tables 4.1, 4.2 and 4.3, we note that with a standard set of parameters, VQGAN and StyleGAN are much better for privacy

than Fawkes. In order to match Fawkes privacy results, we need to change the VQGAN and StyleGAN parameters tenfold. While these parameter changes decrease privacy significantly, generative methods still have a disruptive effect on image quality.

It is worth noting that optimising generative methods using a single embedding method is insufficient for adequate facial image privacy protection. In the case of transferring images to MagFace (Table 4.3), the correct identity for the modified image is often among the top 5 possibilities. Thus, in the next subsection, we use two different embedding methods in the optimisation process to generate private image versions.

Furthermore, we have observed that combining Fawkes poisoning with our proposed methods can be advantageous for facial image privacy protection. We expand on that in Appendix C.

4.6.2 Experiment 2: Comparing StyleGAN and VQGAN Optimised with 2 Embedding Methods on the LFW Dataset

We now compare standard versions of VQGAN and StyleGAN together with other specific versions: StyleGAN_0.02_500 and VQGAN_0.04_128, StyleGAN_0.02_1000 and VQGAN_0.03_512.

These versions are proposed so that they have similar transfer recall scores in each pair. Specifically, average transfer recall scores for different methods are:

- 20.12% for StyleGAN,
- 32.33% for StyleGAN_0.02_500,
- 36.23% for StyleGAN_0.02_1000,
- 42.41% for VQGAN,
- 36.99% for VQGAN_0.03_512 and
- 33.07% for VQGAN_0.04_128.

We create these specific versions of VQGAN and StyleGAN so that we can fairly compare the quality of the generated private images produced by the generative methods. We want to know which method produces the best image quality for a given threshold of our privacy metric.

An example of an evaluation result without transfer is presented in Table 4.4 for MagFace and in 4.5 for MobileFaceNet.

We also study the transfer to different embedding methods. One of the results of this study (for the embedding method SphereFace) is presented in Table 4.6. Transfer results for other embedding methods can be found in the Appendix C.

Figure 4.3: Experiment 2: Examples of images from the LFW dataset modified by different privacy methods: StyleGAN, StyleGAN_0.02_500, StyleGAN_0.02_1000, VQGAN, VQGAN_0.03_512, VQGAN_0.04_128 with embedding methods MagFace and MobileFaceNet. Original and modified Fawkes image versions can be seen in Fig. 4.2. These images have different privacy levels and different qualities: the human rating experiment performs comparisons between images produced by methods with similar recall, *i.e.*, similar privacy results.

From the table 4.6 compared to 4.2 we can see that, in general, generative methods optimised with two embeddings transfer better to other embedding methods than generative methods optimised with only one embedding. For instance, for the unused in an optimisation process embedding SphereFace, the percentage score for StyleGAN with standard parameters optimised with 2 different embedding methods is more than 15 while it was

just around 5 for one embedding method. Examples of images produced by the methods of experiment 2 are presented in Fig. 4.3. More of the examples can be found in the Appendix C.

4.6.3 Experiment 3: Comparing StyleGAN, VQGAN, and Fawkes on the CC dataset

Here we choose the specific versions of VQGAN, StyleGAN, and Fawkes that have similar transfer recall scores in each group for the dataset CC:

- VQGAN_0.003_128 and Fawkes;
- StyleGAN_0.02_1000 and VQGAN_0.04_4096.

In addition, we have compared our results to those obtained with AMT-GAN. However, it is important to note that a direct comparison between our proposed methods and AMT-GAN is not possible, as AMT-GAN is unable to transfer makeup to faces that were not detected. Therefore, in cases where faces were not detected, we had to replace them with the original images, which may affect the comparability of the results. Transfer recalls for the proposed methods are the following:

- AMT-GAN: 49.57%,
- Fawkes: 79.21%,
- StyleGAN_0.02_1000: 24.71%,
- VQGAN_0.003_128: 74.76%,
- VQGAN_0.04_4096: 26.09%,
- VQGAN: 40.45%.

We compare how well proposed generative and pixel-based approaches protect privacy against different embedding methods. One of the results of this study (for the embedding method SphereFace) is presented in Table 4.7. Transfer results for other embedding methods can be found in the Appendix C. Examples of images produced by the methods of experiment 3 are presented in Fig. 4.4. More examples can be found in the Appendix C.

Using table 4.7, we can conclude that, despite using the CC dataset instead of LFW, generative methods prove to be effective for privacy preservation and tend to outperform both the pixel-based method Fawkes and the generative makeup transfer method AMT-GAN.

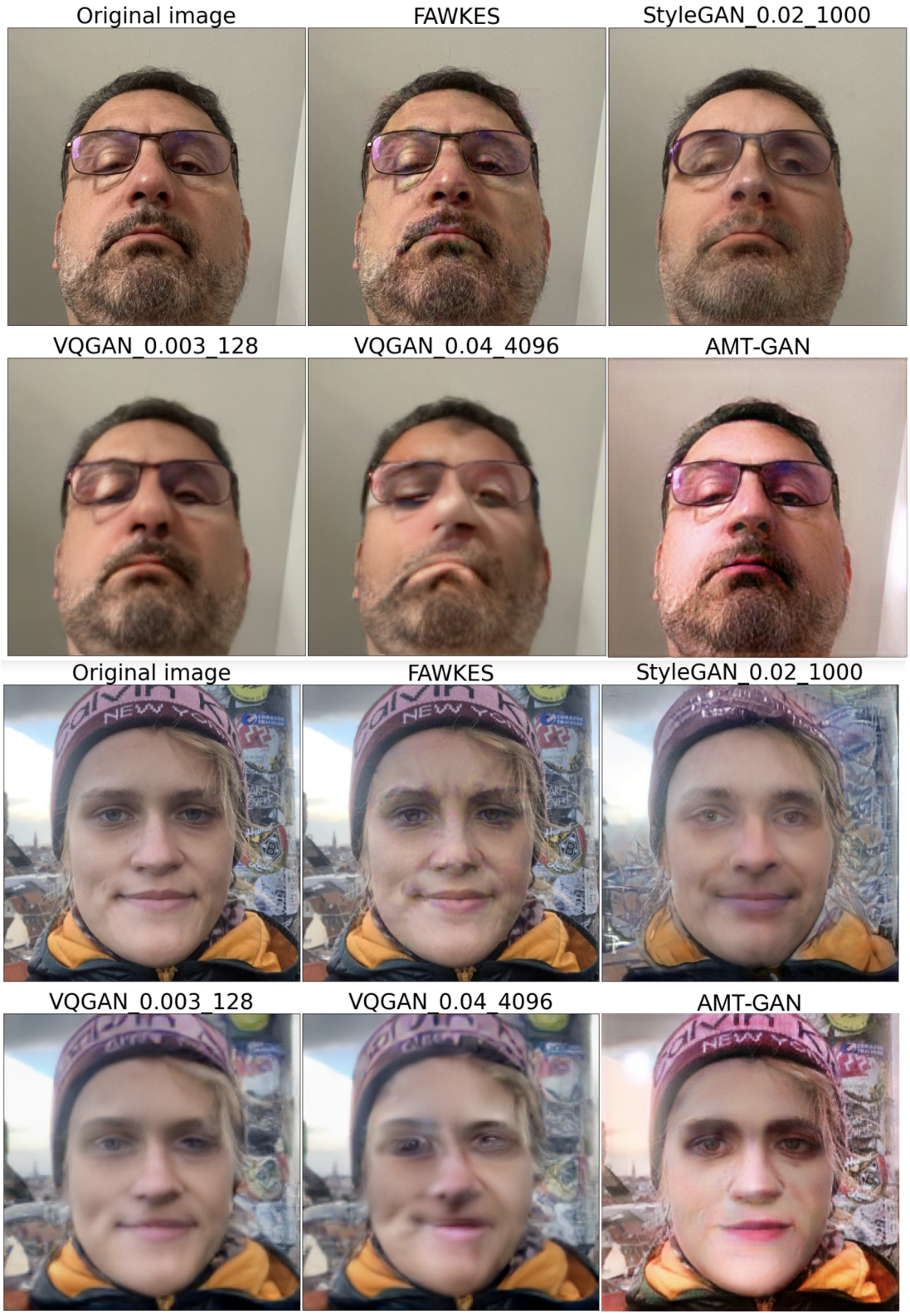

Figure 4.4: Experiment 3: Examples of images of volunteers modified by various privacy methods, including AMT-GAN, Fawkes and StyleGAN_0.02_1000, VQGAN_0.003_128, VQGAN_0.04_4096 optimised with embedding methods MagFace and MobileFaceNet. The different methods and parametrizations lead to different image quality/privacy results; the human rating experiments will compare the quality for methods with similar recall.

4.6.4 Human preferences for similar transfer recall

In Figs 4.4 and 4.3, we can see that, in some cases, modifying the number of iterations in optimisation and the coefficient K affects the quality of an image and its privacy protection. Therefore, to evaluate the modified image quality for different privacy methods with similar transfer recall, we conducted a human preference study. Three human raters were presented with 40 pairs of images generated by the methods discussed in section 4.6.3. Given two images generated by two different methods, the human rater could choose

- "I prefer the left one as an avatar,"
- "I prefer the right one as an avatar," or
- "No preference."

To provide context, the human raters involved in the experiment were not paid and were not authors of the experiment. They were selected using the snowball principle, and their task was to assess the quality and similarity of the modified image to its original version. Without this assessment, we could end up with a black square instead of a privacy-protected image.

The human raters did not have a degree in computer science and were not informed that the experiment related to privacy. However, the instructions provided to them, which included presenting the original image at the centre and emphasising that all images were reasonably close to it, as well as providing examples of potential use cases such as social networks, news articles, and dating websites, made it clear that assessing image similarity was an integral part of the task.

The human-assessment results are presented in Table 4.8.

We compared 5 different pairs of privacy-preserving methods with similar target recall values. In the low privacy (high transfer recall) setting for both LFW and CC datasets, we were able to compare, in terms of quality, the Fawkes method with generative methods specifically modified to match Fawkes transfer recall values.

When we choose FaceNet as an embedding for generative methods (StyleGAN) optimisation, the quality of the images generated by StyleGAN appears to be worse than that of Fawkes. However, when we use MagFace and MobileFaceNet as embeddings for generative methods optimisation, we obtain similar image quality results for VQGAN and Fawkes (51.5%$\pm$3.06%), while VQGAN has a better transfer recall (74.76% compared to 79.21%).

In the high privacy (low transfer recall) setting for both the LFW and CC datasets, we were able to compare different modifications of the generative methods (VQGAN and StyleGAN) with similar transfer recall. In every case, it appears that human raters preferred VQGAN-generated images over StyleGAN-generated images.

4.6.5 Human Identification of Same-Person Images

In this section, our objective is to evaluate the effectiveness of various privacy preservation methods for their applicability on social media platforms and to determine the extent to which people can identify the person after privacy-preservation modifications by different methods, namely:

- VQGAN_0.005_128,
- AMT-GAN, and
- Fawkes, and
- the anonymization method Deep Privacy 2 [HL23].

The experiment is structured as follows: we begin with the original image, referred to as the "original" in the filename. We then examine privacy-preserved versions of different images of the same individual, followed by random images of different people. The central question for each image in a given set is, "Is this the same person as in the original?".

To execute this experiment, we established a setup illustrated in Figure 4.5. This schema visually represents the process, illustrating the different image types involved in the human identification experiment.

We recruited 5 human evaluators, aged from 15 to 43 years, to assess pairs of images, comprising the original image and constructed using privacy-preserving methods for the same individual, as outlined in the schema in Figure 4.5. The results of the human study are presented in Table 4.9.

From the findings presented in Table 4.9, we deduce that VQGAN_0.005_128, AMT-GAN, and Fawkes generate images that are similarly identifiable by human evaluators, with approximately 80% of images generated by these methods being successfully recognised as the same as the original.

Conversely, the anonymization method Deep Privacy 2 frequently produces images that cannot be identified as those of the same individual. This further substantiates that while anonymization methods such as Deep Privacy 2 preserve certain attributes of images, they may fail to preserve their utility.

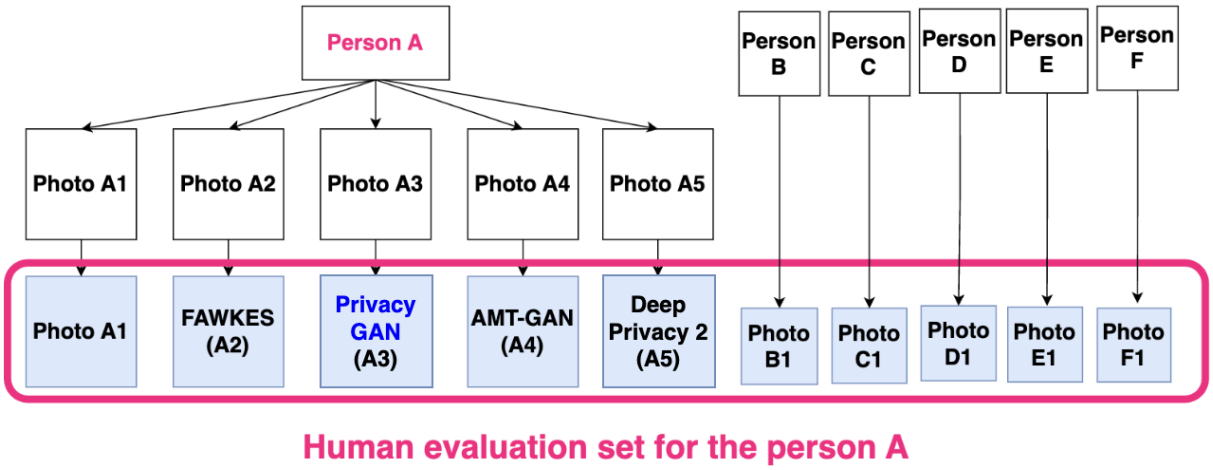

Figure 4.5: Schema of the human identification experiment setup. The experiment involves comparing the original image with privacy-preserved versions of images of the same person and random images of different people to determine whether privacy-preservation methods could preserve the utility of the images modified by them: we expect human raters to recognize the original face for privacy-preserving methods, and not for anonymization methods.

In contrast, our method, along with other utility-focused methods, while not providing absolute protection against facial recognition, effectively safeguards human facial images against prevalent face recognition techniques and maintains image utility for social media use.

This experiment was designed specifically to showcase that the recognisability of our method, along with AMT-GAN and Fawkes, is higher than that of anonymisation methods such as Deep Privacy 2. This explains our focus on testing just one version of PrivacyGAN.

In future experiments, we intend to incorporate multiple versions of PrivacyGAN, and to involve a larger pool of human raters. This approach aims to determine which versions of PrivacyGAN perform best in terms of privacy/recall balance, allowing us to construct a more comprehensive Pareto front for evaluation and comparison.

4.7 Limitations and Future Work

While generative methods are effective in safeguarding image privacy against various embedding methods, they cannot be compared to anonymization techniques. As [RDT21] argues, it is always possible for new recognition attacks to be effective against provided data poisoning methods. However, our approach is different from anonymization, as we do not aim to

provide a privacy guarantee against future attacks. Instead, our objective is to protect users against stalkers and unauthorized identification using current state-of-the-art recognition methods, while still enabling them to share their photos online, and be recognised by their family and friends.

In the future, we would also like to use face enhancement as a tool for or against privacy methods, and check how much PrivacyGAN can be combined with AMT-GAN.

4.8 Conclusions

In conclusion, our contributions in this chapter are

- A new approach to privacy based on inspirational generation, namely PrivacyGAN, using generative models for generating faces close to a given target. This method is orthogonal to the principles of AMT-GAN, so that our method could be used as a first step before AMT-GAN;
- A comparison between our proposed method PrivacyGAN and traditional pixel-based methods, including transfer to unknown embeddings (a.k.a. robustness to unknown embeddings used for identifying people) and human raters for validating image quality;
- A new privacy evaluation method based on the percentage of dataset images that are closer in an embedding space to a modified "private" image than to an original image;
- A new dataset that includes facial images extracted from the Casual Conversations videos.

At the end, we recommend generative methods (Alg. 5), with several embeddings so that robustness and transfer to new methods are properly tested.

According to the human ratings study, Fawkes might be better than StyleGAN for generating high-quality images in the category "low privacy" (recall rate of 90%) on LFW. However, VQGAN and Fawkes have similar results in a low-privacy (74.76-79.21% as a transfer recall) setting, while VQGAN provides better privacy protection.

Among the proposed generative methods, VQGAN is better than StyleGAN overall in terms of quality for a given privacy threshold (see Table 4.8). By the human identification study (section 4.6.5) we further show that, in contrast to anonymization methods, our method, along with other

utility-focused methods, effectively safeguards facial image privacy against prevalent face recognition techniques while maintaining image utility for social media use.

In comparison to AMT-GAN, our method demonstrates superior privacy outcomes depending on the parameter settings, although it doesn't necessarily enhance human recognizability. While AMT-GAN excels in scenarios where recognizability is high and privacy is low, PrivacyGAN offers a broader spectrum of applications, particularly in cases where definitive facial detection is challenging, or where makeup contradicts the user's personal beliefs.

	PrivacyGAN				Pixel-based
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes
Percentage	8.110	0.654	**14.696**	0.861	0.782
Recall@1: m.i.	1.754	22.085	**0.047**	19.242	20.521
Recall@1: o.i.	2.180	22.133	**0.332**	22.180	23.886
Recall@3: m.i.	6.114	61.564	**0.758**	54.597	56.398
Recall@3: o.i.	5.308	60.142	**0.900**	53.981	58.246
Recall@5: m.i.	8.815	77.678	**1.374**	70.711	74.502
Recall@5: o.i.	7.109	75.782	**1.327**	67.820	72.796
Recall@10: m.i.	13.365	86.256	**2.986**	79.668	83.412
Recall@10: o.i.	11.422	85.355	**2.512**	77.773	82.938
Recall@50: m.i.	32.417	94.408	**11.280**	92.227	92.986
Recall@50: o.i.	28.768	94.028	**10**	92.464	93.981
Recall@100: m.i.	43.697	96.303	**19.336**	95.166	95.592
Recall@100: o.i.	40.806	96.019	**17.062**	95.071	96.303

Table 4.1: Test on the LFW dataset. Evaluation for the same embedding that was used for training (no transfer): PrivacyGAN (based on VQGAN or StyleGAN) is optimised with FaceNet, tested with FaceNet, and compared to Fawkes in "high" mode (meaning: high privacy). We see that PrivacyGAN equipped with standard versions of StyleGAN and VQGAN obtains better privacy results compared to Fawkes.

	PrivacyGAN				Pixel-based
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes
Percentage	**5.181**	1.388	5.104	1.271	1.213
Recall@1: m.i.	9.526	21.896	**8.768**	21.043	21.611
Recall@1: o.i.	9.431	22.891	**8.578**	23.223	23.791
Recall@3: m.i.	21.422	53.744	**20.142**	54.929	55.877
Recall@3: o.i.	19.621	53.744	**17.678**	54.360	56.588
Recall@5: m.i.	27.915	67.109	**26.066**	68.294	69.668
Recall@5: o.i.	24.834	66.682	**23.128**	66.777	69.100
Recall@10: m.i.	35.261	74.739	**33.175**	76.161	77.014
Recall@10: o.i.	32.322	75.308	**30.521**	74.455	77.393
Recall@50: m.i.	55.592	86.493	**53.602**	87.867	87.536
Recall@50: o.i.	53.507	87.773	**51.422**	86.967	88.673
Recall@100: m.i.	65.308	91.232	**63.697**	91.469	91.754
Recall@100: o.i.	63.744	91.896	**61.327**	91.611	91.943

Table 4.2: Evaluation in the case of transfer to another embedding on the LFW dataset: PrivacyGAN (with VQGAN or StyleGAN) are optimised with FaceNet and tested with SphereFace. Generative methods do obtain better privacy results than Fawkes, except for the versions specifically created (weakened) to have privacy results similar to Fawkes (these versions are created for comparing image quality in Table 4.8 in a context with equal privacy performance).

	PrivacyGAN				Pixel-based
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes
Percentage	**0.767**	0.361	0.627	0.422	0.408
Recall@1: m.i.	**20.758**	24.028	24.028	24.265	24.882
Recall@1: o.i.	**21.611**	25.972	22.701	25.545	25.403
Recall@3: m.i.	**60.237**	71.848	66.493	73.744	75.403
Recall@3: o.i.	**60.142**	73.602	65.261	73.507	73.507
Recall@5: m.i.	**78.294**	96.967	86.209	98.389	98.768
Recall@5: o.i.	**76.777**	97.156	83.744	98.436	98.863
Recall@10: m.i.	**85.687**	97.962	91.043	98.863	99.194
Recall@10: o.i.	**84.313**	98.152	89.431	98.910	99.005
Recall@50: m.i.	**93.223**	99.005	95.545	99.005	99.194
Recall@50: o.i.	**92.512**	98.957	95.450	99.052	99.194
Recall@100: m.i.	**95.308**	99.052	97.062	99.100	99.194
Recall@100: o.i.	**94.976**	99.052	96.825	99.147	99.194

Table 4.3: Evaluation on the LFW dataset in the case of transfer to another embedding: PrivacyGAN (with VQGAN or StyleGAN) is optimised with FaceNet and tested with MagFace. Generative methods do obtain better privacy results than Fawkes, except for the versions specifically created (weakened) to have privacy results similar to Fawkes (these versions are created for comparing image quality in Table 4.8 in a context with equal privacy performance). However, both Fawkes and generative methods optimised with one embedding do not transfer well to the novel embedding methods such as MagFace, while they transfer better to some other embedding methods such as SphereFace, as in Table 4.2.

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512	VQGAN _0.04_128
Percentage	**15.049**	7.909	7.264	4.910	7.472	8.307
Recall@1: m.i.	**0.095**	0.806	0.900	0.521	0.284	**0.095**
Recall@1: o.i.	**3.555**	8.768	10.521	11.185	6.588	6.919
Recall@3: m.i.	**3.270**	10.711	11.327	15.071	9.147	8.057
Recall@3: o.i.	**6.493**	17.583	20.332	23.981	14.360	14.218
Recall@5: m.i.	**5.118**	16.919	19.479	26.682	17.583	14.834
Recall@5: o.i.	**8.863**	21.943	25.071	30.664	19.147	17.867
Recall@10: m.i.	**9.242**	25.261	28.057	37.488	24.787	22.180
Recall@10: o.i.	**12.275**	28.626	32.133	40	25.972	24.929
Recall@50: m.i.	**23.649**	44.550	46.919	57.678	44.692	42.085
Recall@50: o.i.	**26.114**	48.436	50.521	60.806	46.967	44.028
Recall@100: m.i.	**31.706**	53.555	57.204	66.730	54.597	51.991
Recall@100: o.i.	**33.649**	57.393	60.332	69.052	55.450	54.028

Table 4.4: Evaluation of various PrivacyGAN variants on the LFW dataset, case without transfer: PrivacyGAN (equipped with VQGAN and StyleGAN, including variants) are optimised with MagFace and MobileFaceNet and tested with MagFace. Lower recall means better privacy. Compared to Fawkes, results in Table 4.3: generative methods do get better privacy results. However, we did use MagFace in the algorithm, whereas Fawkes does not, hence the need for further validation (*i.e.*, testing in the case of transfer to embeddings not used in the privacy algorithm), which is done, for example, in Table 4.6.

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512	VQGAN _0.04_128
Percentage	**6.925**	3.290	3.079	2.583	3.555	4.461
Recall@1: m.i.	**0.569**	2.986	2.749	2.227	0.900	0.664
Recall@1: o.i.	**6.635**	14.739	14.976	15.782	12.654	10.379
Recall@3: m.i.	**11.090**	22.417	24.408	26.066	20.190	17.062
Recall@3: o.i.	**13.365**	30.095	32.180	34.834	26.351	22.085
Recall@5: m.i.	**18.531**	36.635	38.152	42.607	31.517	28.246
Recall@5: o.i.	**18.009**	38.578	40.379	45.308	34.550	29.289
Recall@10: m.i.	**26.682**	48.389	49.953	55.735	43.223	39.052
Recall@10: o.i.	**24.739**	48.720	49.100	54.739	44.645	38.720
Recall@50: m.i.	**47.014**	69.384	71.611	74.171	65.687	59.668
Recall@50: o.i.	**46.872**	69.716	71.991	74.597	66.919	60.427
Recall@100: m.i.	**56.682**	77.583	78.957	80.237	73.981	68.578
Recall@100: o.i.	**56.967**	76.588	79.336	80.948	75.308	69.431

Table 4.5: Evaluation on LFW dataset: VQGAN and StyleGAN (and their variants) are optimised with MagFace and MobileFaceNet, and recognition is tested with MobileFaceNet. Generative methods do obtain better privacy than Fawkes (Table C.2).

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512
Percentage	**12.273**	8.157	7.715	6.211	7.387
Recall@1: m.i.	**2.749**	5.118	5.735	6.682	4.787
Recall@1: o.i.	**5.261**	7.536	9.431	10.047	7.204
Recall@3: m.i.	**6.256**	11.801	12.701	14.408	12.512
Recall@3: o.i.	**9.573**	14.218	18.199	17.630	14.408
Recall@5: m.i.	**8.578**	15.924	17.109	19.289	17.536
Recall@5: o.i.	**11.848**	17.725	22.227	22.464	19.005
Recall@10: m.i.	**13.033**	20.711	22.891	26.398	24.360
Recall@10: o.i.	**15.877**	24.218	27.441	29.858	26.303
Recall@50: m.i.	**26.209**	39.858	41.943	47.441	42.749
Recall@50: o.i.	**30.379**	43.554	45.403	51.659	46.161
Recall@100: m.i.	**35.024**	49.336	50.995	57.820	52.749
Recall@100: o.i.	**39.716**	52.654	55.024	61.754	56.256

Table 4.6: Evaluation of various PrivacyGAN variants in the case of transfer to another embedding on the LFW dataset: PrivacyGAN equipped with VQGAN or StyleGAN is optimised with MagFace and MobileFaceNet and tested with SphereFace. In this case, generative methods optimised with two embeddings obtain better results than generative methods optimised with only one embedding method in Table 4.2.

	PrivacyGAN			Pixel-	PrivacyGAN	Adversarial
	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	based Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	9.424	13.399	16.379	7.519	**17.024**	14.067
Recall @1: m.i.	17.854	9.529	5.998	23.952	**5.416**	9.328
Recall @1: o.i.	18.034	11.214	**6.800**	24.092	6.841	8.445
Recall @3: m.i.	40.702	21.364	13.561	52.979	**12.197**	19.980
Recall @3: o.i.	41.765	23.149	14.483	53.420	**14.443**	17.593
Recall @5: m.i.	48.245	26.439	17.051	61.244	**15.727**	24.293
Recall @5: o.i.	49.629	27.924	**17.994**	61.224	18.134	21.344
Recall @10: m.i.	53.220	31.515	21.143	65.055	**20.100**	29.228
Recall @10: o.i.	54.945	33.621	**22.768**	65.135	23.531	25.436
Recall @50: m.i.	64.253	44.835	34.343	72.979	**32.618**	42.086
Recall @50: o.i.	65.537	47.442	**36.068**	72.738	36.911	36.409
Recall @100: m.i.	68.726	51.675	41.484	76.108	**39.478**	49.509
Recall @100: o.i.	70.030	54.363	44.152	75.928	45.176	**42.467**

Table 4.7: Evaluation in the case of a transfer to another embedding on the CC dataset: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet and tested with SphereFace. PrivacyGAN basically outperforms Fawkes while the comparison with AMT-GAN (which could be used on top of our method) depends on criteria and parameters.

Transfer recall	Avatar method 1	Avatar method 2	Human preference: success rate of 1 vs 2
	High privacy (low recall), LFW dataset		
32.6%	(StyleGAN _0.02_500, MagFace + MobileFaceNet)	**(VQGAN (_128_0.04, MagFace + MobileFaceNet)**	43.75 ±5%
36.7%	(StyleGAN _0.02_1000, MagFace + MobileFaceNet)	**(VQGAN _0.03_512, MagFace + MobileFaceNet)**	35.37% ± 5%
	Low privacy (high recall), LFW dataset		
90%	**(StyleGAN _0.003_500 FaceNet)**	Fawkes	87.2% ± 2%
	High privacy, (low recall), CC dataset		
26.09% - 24.71%	**(VQGAN _0.04_4096, MagFace + MobileFaceNet)**	(StyleGAN _0.02_1000, MagFace + MobileFaceNet)	55.2 % ± 3.59
	Low privacy (high recall), CC dataset		
74.76% - 79.21%	**(VQGAN _0.003_128) MagFace + MobileFaceNet)**	Fawkes	51.5 % % % % ± 3.06

Table 4.8: We modify the strength of different privacy-protection-algorithm perturbations until we get to similar target recall levels. We compare the quality of images, for each recall level. Text in bold font refers to human preference, for each recall level (see rightmost column).

Model	Accuracy for humans	99% confidence interval
PrivacyGAN using VQGAN_0.005_128	0.796	± 0.04
AMT-GAN	0.829	± 0.038
Fawkes	0.842	± 0.037
Deep Privacy 2	0.187	± 0.039

Table 4.9: Human face identification for various privacy preservation models. The table demonstrates that, while our method VQGAN_0.005_128, together with AMT-GAN and Fawkes, generate recognisable images, the anonymization method Deep Privacy 2 often produces images that cannot be recognised as the same person. The purpose and limitations of this experiment are further discussed in the text.

5

Conclusions and Future Work

5.1 Summary of Key Findings

In summary, our research has brought forth the insights into ethical considerations for generative modeling and innovative methodologies. The following detailed conclusions encapsulate the contributions and advancements made in each aspect.

5.1.1 Chapter 2: Fairness in Generative Modeling

- **Algorithmic Bias Correction:**

  We proposed an algorithm that utilizes unsupervised reweighting, improving fairness even when the sensitive variables are unknown.

- **Strategies for Addressing Bias Introduced by Image Quality Improvement Techniques:**

  We evaluated image generation techniques like EvolGAN, which utilize image quality estimators to enhance the realism of the produced images. Our findings revealed that such techniques, while improving image quality, also tend to magnify biases present in the training data. Our proposed bias mitigation methods demonstrated heightened effectiveness compared to existing approaches, addressing image quality and fairness simultaneously.

5.1.2 Chapter 3: Enhancing Image Diversity

- **Unsupervised Diversity Technique:**

We introduced the Diverse Diffusion algorithm that enhances image diversity in text-to-image models by expanding the distances between points in a latent space. Our method works in an unsupervised fashion and outperforms state-of-the-art meta-prompt methods, ensuring increased diversity and richness in generated images.

- **Experimental Validation for Underrepresented Categories:**

  We provided experimental validation of our proposed techniques applied to Stable Diffusion on a variety of prompts and multiple metrics such as LPIPS, color diversity evaluation, and gender/ethnicity evaluation. We demonstrated superior performance in representing underrepresented categories, addressing diversity gaps evident in Stable Diffusion.

5.1.3 Chapter 4: Privacy-Preserving Generative Models

- **Privacy Preservation with Generative Methods:**

  We proposed the PrivacyGAN algorithm, based on the proximity of dataset images to a modified "private" image in an embedding space. We applied our method to VQGAN and StyleGAN and demonstrated that our method outperforms existing privacy methods like FAWKES by effectively preserving privacy while maintaining image utility.

- **Novel Privacy Evaluation Methodology:**

  We introduced a novel privacy evaluation method, based on the image proximity in an embedding space relative to the percentage of the dataset size. It redefines privacy evaluation criteria, helping to ensure stronger privacy protection guarantees.

- **Creation of a New Public Dataset for Privacy:**

  We created a new, public facial image dataset for privacy evaluation, extracted from Casual Conversations' videos. Unlike others, this dataset presents additional challenges for privacy protection as images may have a similar background and pose, which can be the case with real photos of a person published on different platforms.

5.2 Ethical Considerations

Throughout our work, ethical considerations are woven into the fabric of methodologies and model development.

Our research presents cutting-edge methods, leverages diverse datasets, and consistently outperforms existing benchmarks, contributing to advancements in generative modeling, fairness, diversity enhancement, and privacy preservation within the framework of ethical AI development. The detailed methodologies and datasets introduced pave the way for responsible, impactful, and ethically sound AI advancements.

5.3 Future work

In the dynamic field of Artificial Intelligence, our long-term research agenda is anchored in the pursuit of ethical AI, specifically addressing fairness, diversity, and privacy in generative modeling.

5.3.1 Well-distributed Point Configurations for Generative Modeling Diversity Enhancement

One of our ongoing projects aims to explore optimal point configurations to enhance image diversity for generative models such as stable diffusion [RBL+22] and BigGAN [BDS18].

The proposed research aligns with the findings of one of our papers [ZTN23], where an increase in the distance between latent vectors in Stable Diffusion was explored to enhance diversity in generated images. This work provides a foundation for considering diversity within the latent space of generative models. In future work, we would like to explore a novel direction to this problem: the integration of well-distributed point configurations.

The research endeavors to reshape generative modeling by establishing a novel correlation between well-distributed point configurations and image diversity. This involves exploring supporting mathematics, formulating metrics for point configuration distribution, assessing image diversity for various image generation methods (such as stable diffusion and BigGAN), conducting real-world testing, and integrating the methodology into existing AI frameworks.

Anticipated outcomes include the introduction of new metrics and practical applications, particularly in latent diffusion models.

When implemented, we would like to add our findings to stable diffusion and other text-to-image models, in order to ensure that users have an access to diverse image outputs.

5.3.2 Privacy-Preserving Facial Image Generation

We aim to build on our past work and exploit ideas from PrivacyGAN and the generative makeup method AMT-GAN [HLZ+22] in a new privacy-preservation method. Particularly, we would like to apply a similar optimization strategy (to one in PrivacyGAN) in the latent space of stable diffusion to create realistic and usable images protected from recognition by current neural networks.

This approach will provide another level of control for users, enabling them to set specific modification attributes to their images. While with AMT-GAN, users have to create images with adversarial makeup and with PrivacyGAN, they have to modify the full image, in our proposed method, users will have full control over the features they want to modify for creating privacy-protected versions of their images.

For example, they will be able to specify that they would like to have a particular style of makeup as in VQGAN or adversarial moustache or adversarial glasses on their image.

When implemented, we would like to add privacy feature to majour platforms such as Facebook and Tinder in order to make our research more accessible for the users.

Publications

This part lists all the publications / preprints prepared during the PhD.

- **2023:** Zameshina, M., et al., "Diverse Diffusion: Enhancing Image Diversity in Text-to-Image Generation," *preprint*, [ZTN23].
- **2023:** Zameshina, M., et al., "PrivacyGAN: Robust Generative Image Privacy," *preprint*, [ZCTN23].
- **2022:** Zameshina, M., et al., "Fairness in Generative Modeling: Do It Unsupervised!," *GECCO 2022*, [ZTT+22].
- **2020:** Roziere, B., et al., "EvolGAN: Evolutionary Generative Adversarial Networks," *ACCV 2020*, [RTH+20].

Bibliography

[Ana19] H. Anadon. Face expression and ethnic recognition. https://github.com/HectorAnadon/Face-expression-and-ethnic-recognition, 2019.

[Ata04] Emanouil I Atanassov. On the discrepancy of the halton sequences. *Math. Balkanica (NS)*, 18(1-2):15–32, 2004.

[Bar22] Kyle Barr. Ai image generators routinely display gender and cultural bias. In *gizmodo.com*, 2022.

[BDS18] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *Arxiv preprint*, 1809.11096, 2018.

[Ber22] Sebastian Berns. Increasing the diversity of deep generative models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 12870–12871, 2022.

[BKD+23] Federico Bianchi, Pratyusha Kalluri, Esin Durmus, Faisal Ladhak, Myra Cheng, Debora Nozza, Tatsunori Hashimoto, Dan Jurafsky, James Zou, and Aylin Caliskan. Easily accessible text-to-image generation amplifies demographic stereotypes at large scale. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 1493–1504, 2023.

[BTPS23] Simone Barattin, Christos Tzelepis, Ioannis Patras, and Nicu Sebe. Attribute-preserving face dataset anonymization via latent code optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8001–8010, 2023.

[BYMC22] Hritik Bansal, Da Yin, Masoud Monajatipoor, and Kai-Wei Chang. How well can text-to-image generative models under-

stand ethical natural language interventions? *arXiv preprint arXiv:2210.15230*, 2022.

[CBLC22] Pierre Chambon, Christian Bluethgen, Curtis P Langlotz, and Akshay Chaudhari. Adapting pretrained vision-language foundational models to medical imaging domains. *arXiv preprint arXiv:2210.04133*, 2022.

[CG23] Gilad Cohen and Raja Giryes. Generative adversarial networks. In *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*, pages 375–400. Springer, 2023.

[CGF+21] Valeriia Cherepanova, Micah Goldblum, Harrison Foley, Shiyuan Duan, John Dickerson, Gavin Taylor, and Tom Goldstein. Lowkey: Leveraging adversarial attacks to protect social media users from facial recognition. *arXiv preprint arXiv:2101.07922*, 2021.

[CGS+20] Kristy Choi, Aditya Grover, Trisha Singh, Rui Shu, and Stefano Ermon. Fair generative modeling via weak supervision. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1887–1898. PMLR, 13–18 Jul 2020.

[CLG+23] Hila Chefer, Oran Lang, Mor Geva, Volodymyr Polosukhin, Assaf Shocher, Michal Irani, Inbar Mosseri, and Lior Wolf. The hidden language of diffusion models. *arXiv preprint arXiv:2306.00966*, 2023.

[CLGH18] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, pages 428–438. Springer, 2018.

[CRRN+21] Nathanaël Carraz Rakotonirina, Andry Rasoanaivo, Laurent Najman, Petr Kungurtsev, Jeremy Rapin, Fabien Teytaud, Baptiste Roziere, Olivier Teytaud, Markus Wagner, Pak-Kan Wong, and Vlad Hosu. Many-Objective Optimization for Diverse Image Generation. working paper or preprint, November 2021.

[DGXZ19] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[DL16] Duc-Cuong Dang and Per Kristian Lehre. Self-adaptation of mutation rates in non-elitist populations. In *Parallel Problem Solving from Nature–PPSN XIV: 14th International Conference, Edinburgh, UK, September 17-21, 2016, Proceedings 14*, pages 803–813. Springer, 2016.

[DSDN19] Ali Dabouei, Sobhan Soleymani, Jeremy Dawson, and Nasser Nasrabadi. Fast geometrically-perturbed adversarial faces. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1979–1988. IEEE, 2019.

[ELEM17] Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms. *arXiv preprint arXiv:1706.07068*, 2017.

[ERO21] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[FKN23] Kathleen C Fraser, Svetlana Kiritchenko, and Isar Nejadgholi. Diversity is not a one-way street: Pilot study on ethical interventions for racial bias in text-to-image systems. *ICCV, accepted*, 2023.

[GPAM+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 2014.

[GWT19] Oran Gafni, Lior Wolf, and Yaniv Taigman. Live face de-identification in video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9378–9387, 2019.

[Ham60] J. M. Hammersley. Monte-carlo methods for solving multivariate problems. *Annals of the New York Academy of Sciences*, 86(3):844–874, 1960.

[HAM+19] Eduardo Hargreaves, Claudio Agosti, Daniel Menasché, Giovanni Neglia, Alexandre Reiffers-Masson, and Eitan Altman. Fairness in online social network timelines: Measurements, models and mechanism design. *Performance Evaluation*, 129:15–39, Feb 2019.

[Har22] Drew Harwell. This facial recognition website can turn anyone into a cop—or a stalker. In *Ethics of Data and Analytics*, pages 63–67. Auerbach Publications, 2022.

[HBD+21] Caner Hazirbas, Joanna Bitton, Brian Dolhansky, Jacqueline Pan, Albert Gordo, and Cristian Canton Ferrer. Towards measuring fairness in ai: the casual conversations dataset. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2021.

[HGS19] Vlad Hosu, Bastian Goldlucke, and Dietmar Saupe. Effective aesthetics prediction with multi-level spatially pooled features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9375–9383, 2019.

[HHL+23] Cusuh Ham, James Hays, Jingwan Lu, Krishna Kumar Singh, Zhifei Zhang, and Tobias Hinz. Modulating pretrained diffusion models for multimodal image synthesis. *arXiv preprint arXiv:2302.12764*, 2023.

[HL23] Håkon Hukkelås and Frank Lindseth. Deepprivacy2: Towards realistic full-body anonymization. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1329–1338, 2023.

[HLSS20] Vlad Hosu, Hanhe Lin, Tamas Sziranyi, and Dietmar Saupe. Koniq-10k: An ecologically valid database for deep learning of blind image quality assessment. *IEEE Transactions on Image Processing*, pages 1–1, 2020.

[HLZ+22] Shengshan Hu, Xiaogeng Liu, Yechao Zhang, Minghui Li, Leo Yu Zhang, Hai Jin, and Libing Wu. Protecting facial privacy: generating adversarial identity masks via style-robust

makeup transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15014–15023, 2022.

[HMB+23] Fabio Hellmann, Silvan Mertes, Mohamed Benouis, Alexander Hustinx, Tzung-Chien Hsieh, Cristina Conati, Peter Krawitz, and Elisabeth André. Ganonymization: A gan-based face anonymization framework for preserving emotional expressions. *arXiv preprint arXiv:2305.02143*, 2023.

[HMBLM08] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[HML19] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. Deepprivacy: A generative adversarial network for face anonymization. In *International symposium on visual computing*, pages 565–578. Springer, 2019.

[HO03] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 11(1), 2003.

[Ho23] Cheuk Ting Ho. Stable diffusion: Why are diverse results so hard to come by? In *anaconda.com*, 2023.

[HPK+20] Sunhee Hwang, Sungho Park, Dohyung Kim, Mirae Do, and Hyeran Byun. Fairfacegan: Fairness-aware facial image-to-image translation, 2020.

[HTDX22] Nisha Huang, Fan Tang, Weiming Dong, and Changsheng Xu. Draw your art dream: Diverse digital art synthesis with multimodal guided diffusion. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1085–1094, 2022.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[JKH+21] Ajil Jalal, Sushrut Karmalkar, Jessica Hoffmann, Alex Dimakis, and Eric Price. Fairness for image generation with uncertain sensitive attributes. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4721–4732. PMLR, 18–24 Jul 2021.

[KAH+21] Patrik Joslin Kenfack, Daniil Dmitrievich Arapov, Rasheed Hussain, S. M. Ahsan Kazmi, and Adil Mehmood Khan. On the fairness of generative adversarial networks (gans), 2021.

[KALL18a] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *ICLR*, 2018.

[KALL18b] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[KKSY23] Eunji Kim, Siwon Kim, Chaehun Shin, and Sungroh Yoon. De-stereotyping text-to-image models through prompt tuning. *ICML 2023 Workshop on Deployable GenerativeAI, accepted*, 2023.

[KLA19a] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[KLA19b] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[KLRS17] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[KRMA23] Shyamgopal Karthik, Karsten Roth, Massimiliano Mancini, and Zeynep Akata. If at first you don't succeed, try, try again: Faithful diffusion-based text-to-image generation by selection. *arXiv preprint arXiv:2305.13308*, 2023.

[KY19] Taehoon Kim and Jihoon Yang. Latent-space-level image anonymization with adversarial protector networks. *IEEE Access*, 7:84992–84999, 2019.

[LWY+17] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017.

[MBC79] Michael D. McKay, Richard J. Beckman, and William J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 21:239–245, 1979.

[MDH+20] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models, 2020.

[MTB+22] Vongani H Maluleke, Neerja Thakkar, Tim Brooks, Ethan Weber, Trevor Darrell, Alexei A Efros, Angjoo Kanazawa, and Devin Guillory. Studying bias in gans through the lens of race. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, pages 344–360. Springer, 2022.

[MZHZ21] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14225–14234, 2021.

[Nie92] Harald Niederreiter. *Random Number Generation and quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992.

[OHP+23] David Oniani, Jordan Hilsman, Yifan Peng, Ronald K Poropatich, COL Pamplin, LTC Legault, Yanshan Wang, et al. From military to healthcare: Adopting and expanding ethical principles for generative artificial intelligence. *arXiv preprint arXiv:2308.02448*, 2023.

[Ple21] Max Plenke. The reason this "racist soap dispenser" doesn't work on black skin. https://www.mic.com/articles/124899/the-reason-this-racist-soap-dispenser-doesn-t-work-on-black-skin, 2021.

[PS20] Dana Pessach and Erez Shmueli. Algorithmic fairness, 2020.

[PVZ15] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In Xianghua Xie, Mark W. Jones, and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 41.1–41.12. BMVA Press, September 2015.

[PY00] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *41st Annual Symposium on Foundations of Computer Science*, pages 86–92, 2000.

[QNS+22] Yuying Qiu, Zhiyi Niu, Biao Song, Tinghuai Ma, Abdullah Al-Dhelaan, and Mohammed Al-Dhelaan. A novel generative model for face privacy protection in video surveillance with utility maintenance. *Applied Sciences*, 12(14):6962, 2022.

[RBL+22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[RDT21] Evani Radiya-Dixit and Florian Tramèr. Data poisoning won't save you from facial recognition. *arXiv preprint arXiv:2106.14851*, 2021.

[Riv19] M. Riviere. Pytorch GAN Zoo. https://GitHub.com/FacebookResearch/pytorch_GAN_zoo, 2019.

[RPG+21] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 18–24 Jul 2021.

[RTH+20] Baptiste Rozière, Fabien Teytaud, Vlad Hosu, Hanhe Lin, Jeremy Rapin, Mariia Zameshina, and Olivier Teytaud. EvolGAN: Evolutionary Generative Adversarial Networks. In *Asia Conference on Computer Vision (ACCV)*, Virtual, Japan, November 2020.

[RTR+19] Morgane Riviere, Olivier Teytaud, Jérémy Rapin, Yann LeCun, and Camille Couprie. Inspirational adversarial image generation. *arXiv preprint 1906.11661*, 2019.

[RW18] Eitan Richardson and Yair Weiss. On GANs and GMMs, 2018.

[Sam19] Ian Sample. What is facial recognition-and how sinister is it. *The Guardian*, 29, 2019.

[SAT04] H. Sato, H. E. Aguirre, and K. Tanaka. Local dominance using polar coordinates to enhance multiobjective evolutionary algorithms. In *Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, volume 1, pages 188–195 Vol.1, 2004.

[SBAD+23] Dvir Samuel, Rami Ben-Ari, Nir Darshan, Haggai Maron, and Gal Chechik. Norm-guided latent space exploration for text-to-image generation. *arXiv preprint arXiv:2306.08687*, 2023.

[SBBR16] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pages 1528–1540, 2016.

[SFH+23] Brandon Smith, Miguel Farinha, Siobhan Mackenzie Hall, Hannah Rose Kirk, Aleksandar Shtedritski, and Max Bain. Balancing the picture: Debiasing vision-language datasets with synthetic contrast sets. *arXiv preprint arXiv:2305.15407*, 2023.

[SHCV18] Prasanna Sattigeri, Samuel C. Hoffman, Vijil Chenthamarakshan, and Kush R. Varshney. Fairness gan, 2018.

[SHCV19] P. Sattigeri, S. C. Hoffman, V. Chenthamarakshan, and K. R. Varshney. Fairness gan: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development*, 63(4/5):3:1–3:9, 2019.

[SHK22] Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. The biased artist: Exploiting cultural biases via homoglyphs in text-guided image generation models. *arXiv preprint arXiv:2209.08891*, 2022.

[SJJ19] Joni Salminen, Soon-Gyo Jung, and Bernard J. Jansen. Detecting demographic bias in automatically generated personas. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019. Association for Computing Machinery.

[SKP15] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[SLYZ22] Zeyang Sha, Zheng Li, Ning Yu, and Yang Zhang. De-fake: Detection and attribution of fake images generated by text-to-image diffusion models. *arXiv preprint arXiv:2210.06998*, 2022.

[SNN23] Fahad Shamshad, Muzammal Naseer, and Karthik Nandakumar. Clip2protect: Protecting facial privacy using text-guided makeup via adversarial latent search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20595–20605, 2023.

[SSG+22] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. *arXiv preprint arXiv:2212.03860*, 2022.

[SWT+23] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 769–778, 2023.

[SWZ+20a] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *29th USENIX security symposium (USENIX Security 20)*, pages 1589–1604, 2020.

[SWZ+20b] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *FAWKES github: https://github.com/Shawn-Shan/fawkes*, 2020.

[TC21] Christopher T. H Teo and Ngai-Man Cheung. Measuring fairness in generative models, 2021.

[The23] Danie Theron. Evidence of unfair bias across gender, skin tones & intersectional groups in generated images from stable diffusion. In *towardsdatascience.com*, 2023.

[Tru20] Kevin Truong. This image of a white barack obama is ai's racial bias problem in a nutshell. *vice.com*, 2020.

[TSZ20] Shuhan Tan, Yujun Shen, and Bolei Zhou. Improving the fairness of deep generative models without retraining. *arXiv preprint arXiv:2012.04842*, 2020.

[TYRW14] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[WFL+21] Kuan-Chieh Wang, Yan Fu, Ke Li, Ashish Khisti, Richard Zemel, and Alireza Makhzani. Variational model inversion attacks. *Advances in Neural Information Processing Systems*, 34:9706–9719, 2021.

[WLH+21] Jun Wang, Yinglu Liu, Yibo Hu, Hailin Shi, and Tao Mei. Facex-zoo: A pytorch toolbox for face recognition. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3779–3782, 2021.

[WLVGP09] Thibaut Weise, Hao Li, Luc Van Gool, and Mark Pauly. Face/off: Live facial puppetry. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer animation*, pages 7–16, 2009.

[WSZZ23] Emily Wenger, Shawn Shan, Haitao Zheng, and Ben Y Zhao. Sok: Anti-facial recognition technology. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 864–881. IEEE, 2023.

[XRLS18] Jingjing Xu, Xuancheng Ren, Junyang Lin, and Xu Sun. Diversity-promoting gan: A cross-entropy based generative adversarial network for diversified text generation. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3940–3949, 2018.

[XYZW18] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Fairgan: Fairness-aware generative adversarial networks. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 570–575, 2018.

[YCY17] Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. Midinet: A convolutional generative adversarial network for symbolic-domain music generation. *arXiv preprint arXiv:1703.10847*, 2017.

[ZCTN23] Mariia Zameshina, Marlene Careil, Olivier Teytaud, and Laurent Najman. Privacygan: robust generative image privacy. *arXiv preprint arXiv:2310.12590*, 2023.

[ZIE18a] Richard Zhang, Phillip Isola, and Alexei A Efros. The unreasonable effectiveness of deep features as a perceptual metric.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

[ZIE+18b] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[ZTN23] Mariia Zameshina, Olivier Teytaud, and Laurent Najman. Diverse diffusion: Enhancing image diversity in text-to-image generation. *arXiv preprint arXiv:2310.12583*, 2023.

[ZTT+22] Mariia Zameshina, Olivier Teytaud, Fabien Teytaud, Vlad Hosu, Nathanael Carraz, Laurent Najman, and Markus Wagner. Fairness in generative modeling: do it unsupervised! In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 320–323, 2022.

A

Fairness in generative modeling

Reweighting with respect to four VF binarized variables for a specific target

Table A.1 presents results of different methods in terms of the frequency of black people. In most cases, the frequency of black people decreased from the original 4.8% when applying EvolGan, but increased when applying reweighting. We note exceptions: whereas randomly chosen variables were always beneficial, very correlated variables failed in the most difficult cases.

A.1 Gallery: K512 strata

We check how much K512 succeeds in finding good/bad face generations. We present in Fig. A.1 examples of failed generations, as detected by K512, i.e., the worst 5% generations by StyleGan2: we observe a higher frequency of failed generations with artifacts. This confirms, i.e., that generating with StyleGan2 while biasing by K512 (even by simple filtering) significantly improves the quality of generated images. We present in Fig. A.2 images reaching the top 0.5% of K512 values. Artifacts are rare.

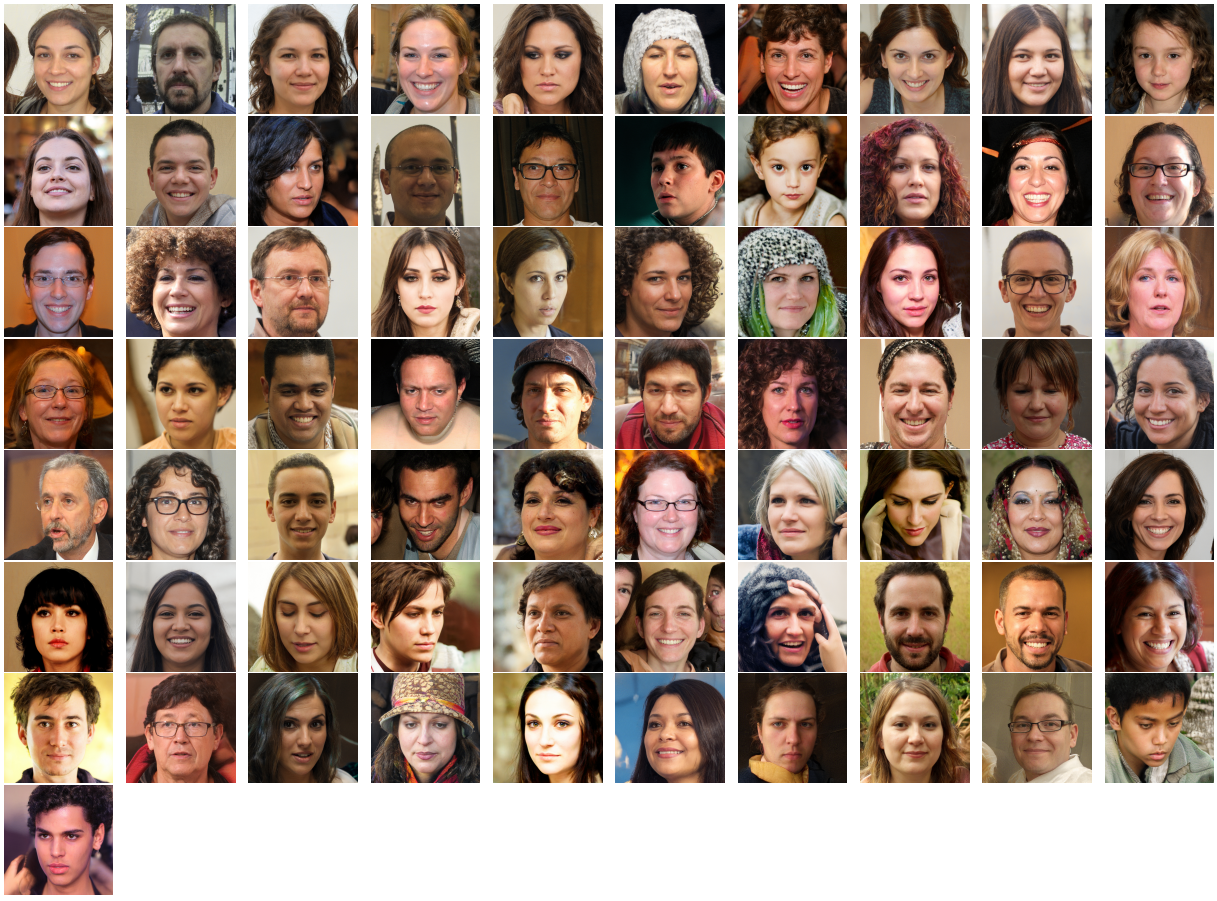

Figure A.1: Failed generations from StyleGan2, according to K512 values. We correctly detect failed generations. See the difference with Fig. A.2.

Figure A.2: Successful generations from StyleGan2, according to K512 (top 5%). Results are good: consistently with [RTH+20], we see that results are ok.

		StyleGan2	EG	Reweigh - EG
Selection rate in EG: 5.1%				
100/1979	corr.	.0480	.0297	.0258
100/1979	random	.0480	.0297	.0309
Selection rate in EG: 7.6%				
150/1979	corr.	.0480	.0271	.0219
150/1979	random	.0480	.0271	.0292
Selection rate in EG: 10.1%				
200/1979	corr.	.0480	.0247	.0207
200/1979	random	.0480	.0247	.0269
Selection rate in EG: 12.6%				
250/1979	corr.	.0480	.0273	.0279
250/1979	random	.0480	.0273	.0277
Selection rate in EG: 15.2%				
300/1979	corr.	.0480	.0239	.0245
300/1979	random	.0480	.0239	.0234
Selection rate in EG: 17.7%				
350/1979	corr.	.0480	.0285	.0333
350/1979	random	.0480	.0285	.0300
Selection rate in EG: 20.2%				
400/1979	corr.	.0480	.0353	.0350
400/1979	random	.0480	.0353	.0374
Selection rate in EG: 22.8%				
450/1979	corr.	.0480	.0358	.0370
450/1979	random	.0480	.0358	.0390
Selection rate in EG: 25.3%				
500/1979	corr.	.0480	.0344	.0354
500/1979	random	.0480	.0344	.0360
Selection rate in EG: 27.8%				
550/1979	corr.	.0480	.0344	.0364
550/1979	random	.0480	.0344	.0363
Selection rate in EG: 25.3%				
600/1979	corr.	.0480	.0331	.0340
600/1979	random	.0480	.0331	.0356

Table A.1: Dataset: faces generated by StyleGan2. The frequency of black people in the different versions, depending on which strata are used for applying the reweighting method of Section 2.3.2. Random: four variables randomly picked up among the 128 binary variables built from VGG-Faces. Correlated: same VGG-Faces, but we use the most correlated ones.

B

Diverse Diffusion: Enhancing Image Diversity in Text-to-Image Generation

B.1 Additional results for color diversity

Here, we provide additional experimental results to further illustrate the color diversity improvement achieved by our proposed approach. The results are organized based on different coefficient values (K) and batch sizes. The figures presented below showcase the multiplicative improvement of batches featuring certain color dominance characteristics using the "pooling_cap" and "pooling_max" methods compared to the baseline Stable Diffusion.

B.1.1 K=1

Batch size = 3

Figures B.1 and B.2 display the results for the $K = 1$ coefficient and a batch size of 3 in both long and standard experiments.

Batch size = 5

Figures B.1 and B.4 present the results for the $K = 1$ coefficient and a batch size of 5 in both long and standard experiments.

Batch size = 10

Figures B.5 and B.6 illustrate the results for the $K = 1$ coefficient and a batch size of 10 in both long and standard experiments.

Batch size = 50

Figure B.7 illustrates the results for the $K = 1$ coefficient and a batch size of 50 in the standard experiment.

B.1.2 K=1.1

Batch size = 3

Figures B.8 and B.9 display the results for the $K = 1.1$ coefficient and a batch size of 3 in both long and standard experiments.

Batch size = 5

Figures B.10 and B.11 present the results for the $K = 1.1$ coefficient and a batch size of 5 in both long and standard experiments.

Batch size = 10

Figures B.12 and B.14 showcase the results for the $K = 1.1$ coefficient and a batch size of 10 in both long and standard experiments.

Batch size = 50

Figure B.15 illustrates the results for the $K = 1$ coefficient and a batch size of 50 in the standard experiment.

B.1.3 K=1.2

Batch size = 3

Figures B.16 and B.17 display the results for the $K = 1.2$ coefficient and a batch size of 3 in both long and standard experiments.

Batch size = 5

Figures B.18 and B.19 present the results for the $K = 1.2$ coefficient and a batch size of 5 in both long and standard experiments.

Batch size = 10

Figures B.13 and B.20 illustrate the results for the $K = 1.2$ coefficient and a batch size of 10 in both long and standard experiments.

Batch size = 50

Figure B.21 illustrates the results for the $K = 1$ coefficient and a batch size of 50 in the standard experiment.

These supplementary results provide an evaluation of the color diversity improvement achieved by our proposed approach across various coefficient values and batch sizes. The figures demonstrate the effectiveness of the "pooling_cap" method in enhancing color diversity compared to the baseline Stable Diffusion method.

B.2 Additional results for LPIPS evaluation

In the main chapter for the small batch sizes, we provide the LPIPS evaluation results only for the "long experiment". Here, we provide missing

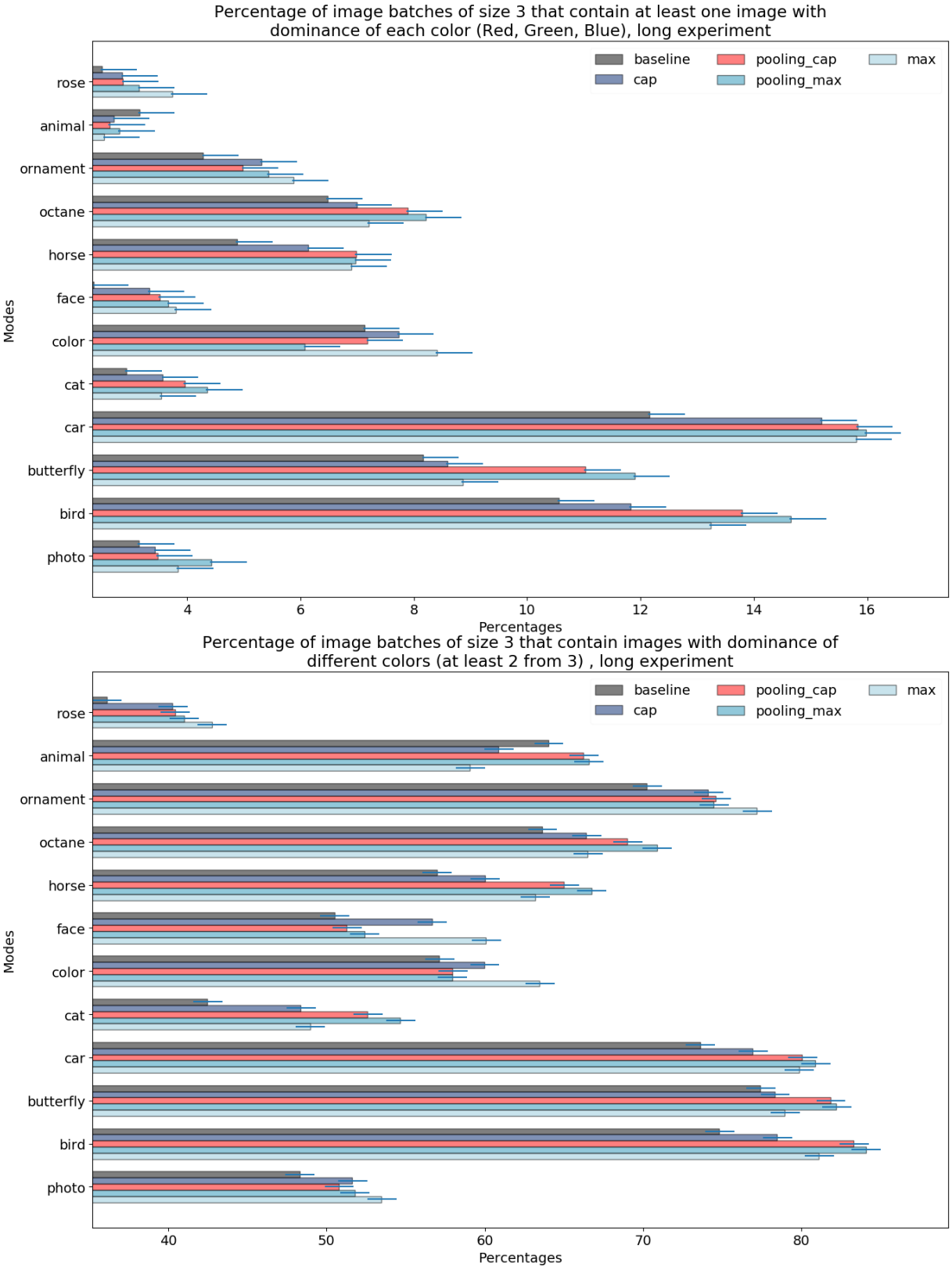

Figure B.1: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 3$, long experiment

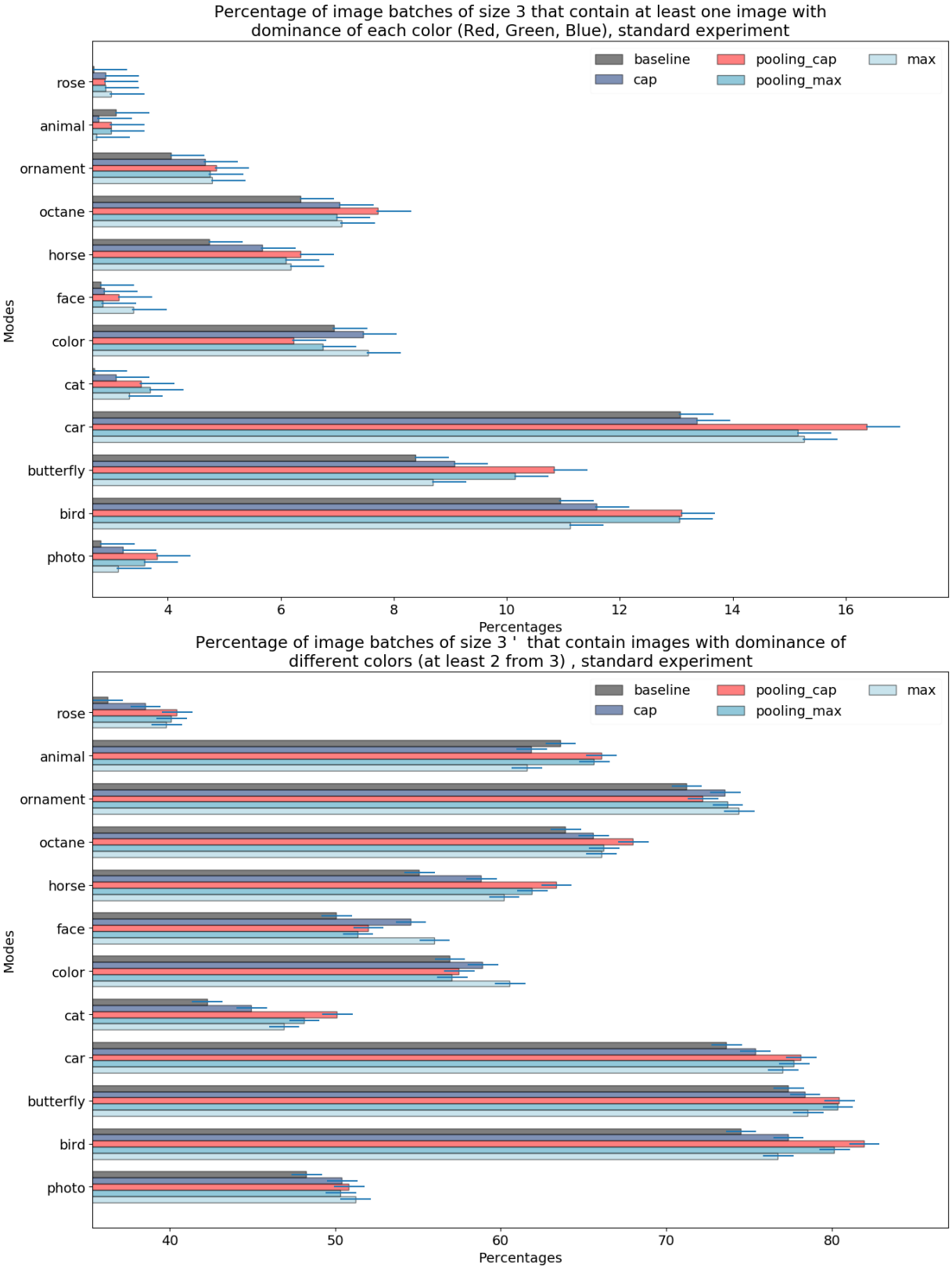

Figure B.2: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 3$, standard experiment

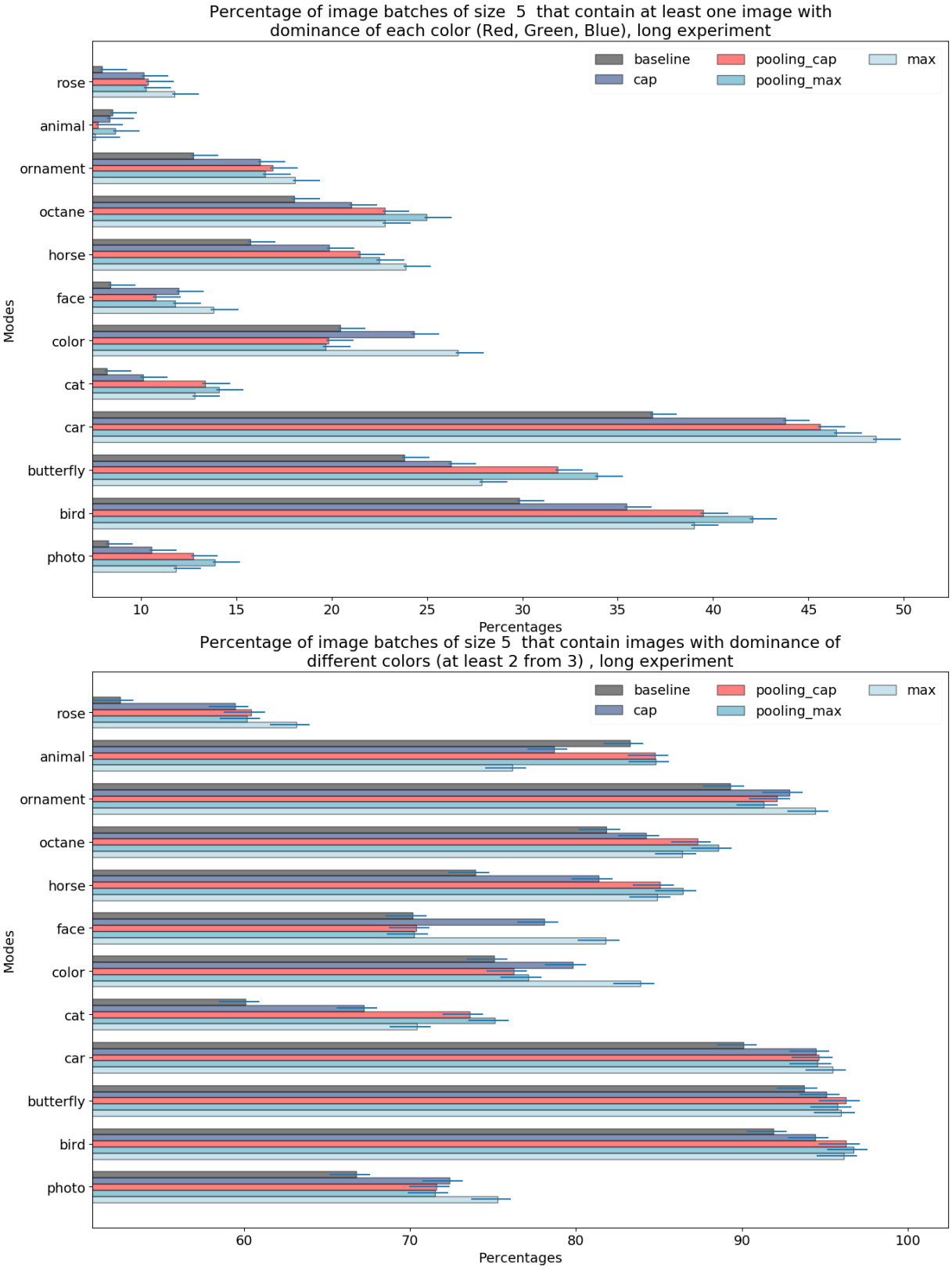

Figure B.3: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 5$, long experiment

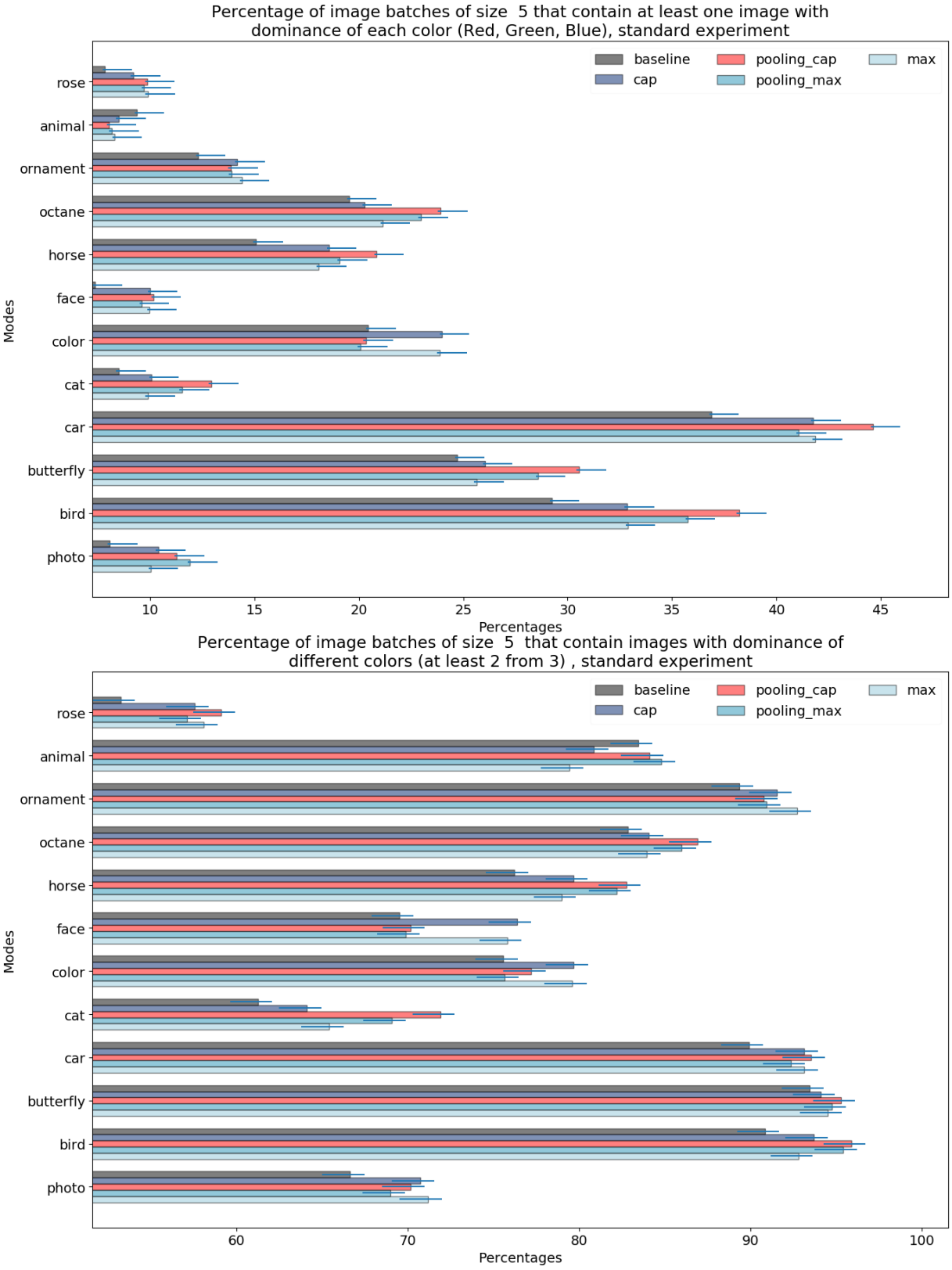

Figure B.4: Multiplicative improvement of batches containing all 3 dominant colors with $K = 1$, batch size $= 5$, standard experiment

results for the "standard experiment" setting.

Figure B.22 presents the average batch pairwise LPIPS distance for a batch size of 3 for the "standard experiment" setting. Here, same as in Figure 3.6 it is hard to see what method is the best for diversity due to the small batch size

Figure B.23 illustrates the average batch pairwise LPIPS distance for a batch size of 5 in the "standard experiment" setting. We can see that the results presented in Figure B.23 are less conclusive than the results in Figure 3.7, which demonstrates that the "long experiment" setting is better suited for the small batch sizes.

Figure B.24 illustrates the average batch pairwise LPIPS distance for a batch size of 10 in the "standard experiment" setting. We can see that similarly to the results for batch size 5, the results presented in Figure B.24 are less conclusive than the results in Figure 3.8, which again demonstrates that the "long experiment" setting is better suited for the small batch sizes.

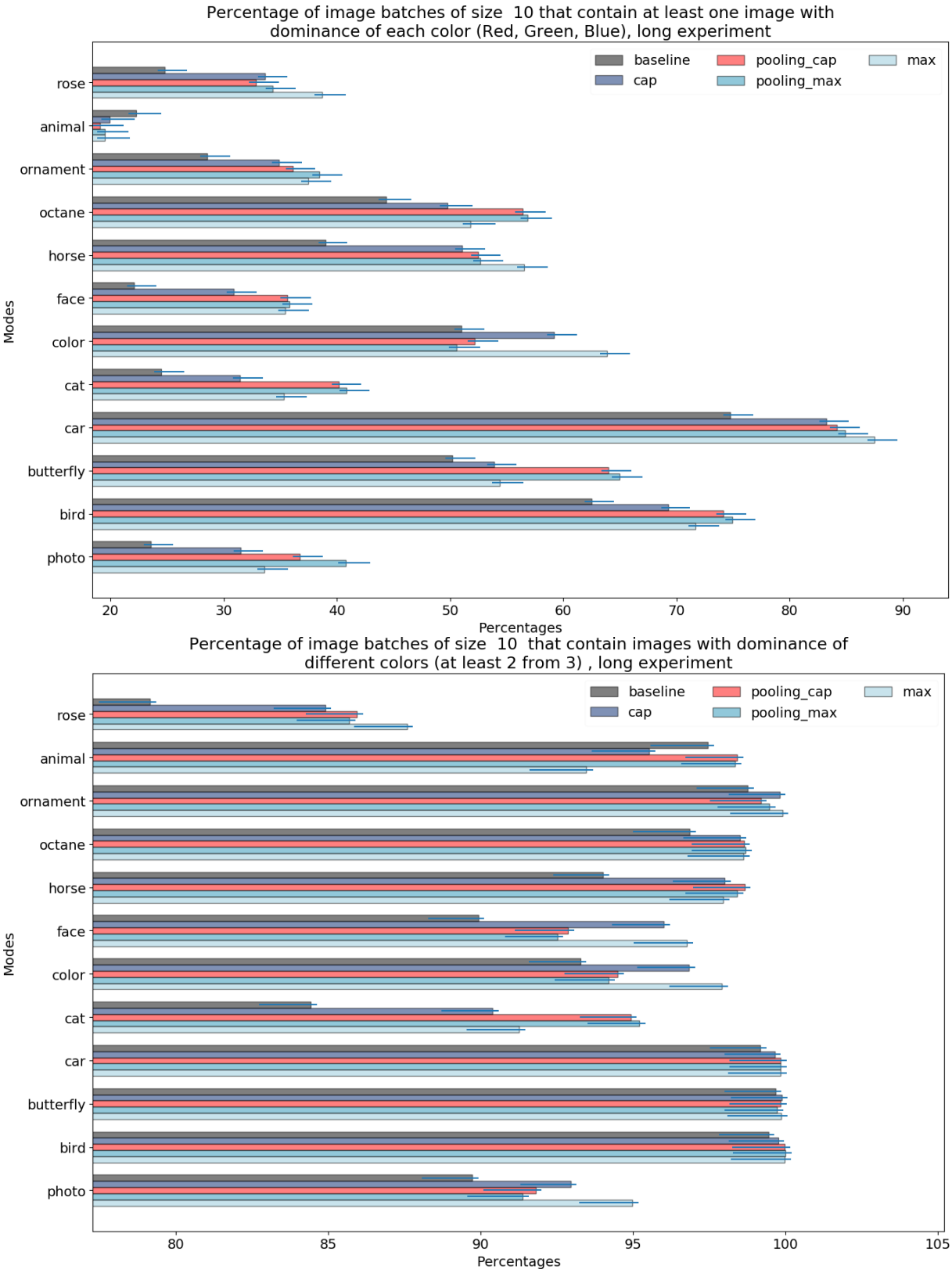

Figure B.5: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 10$, long experiment

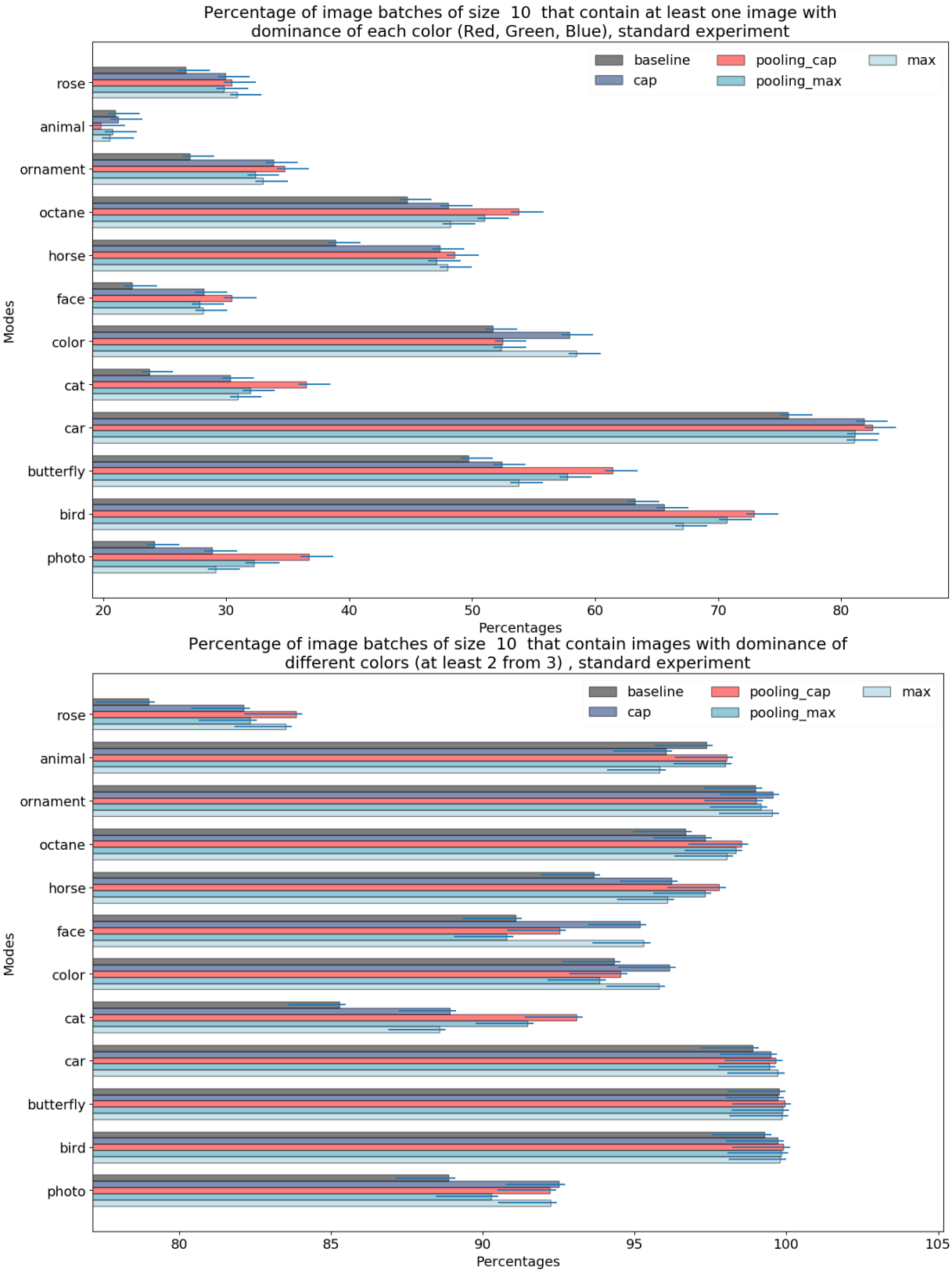

Figure B.6: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 10$, standard experiment

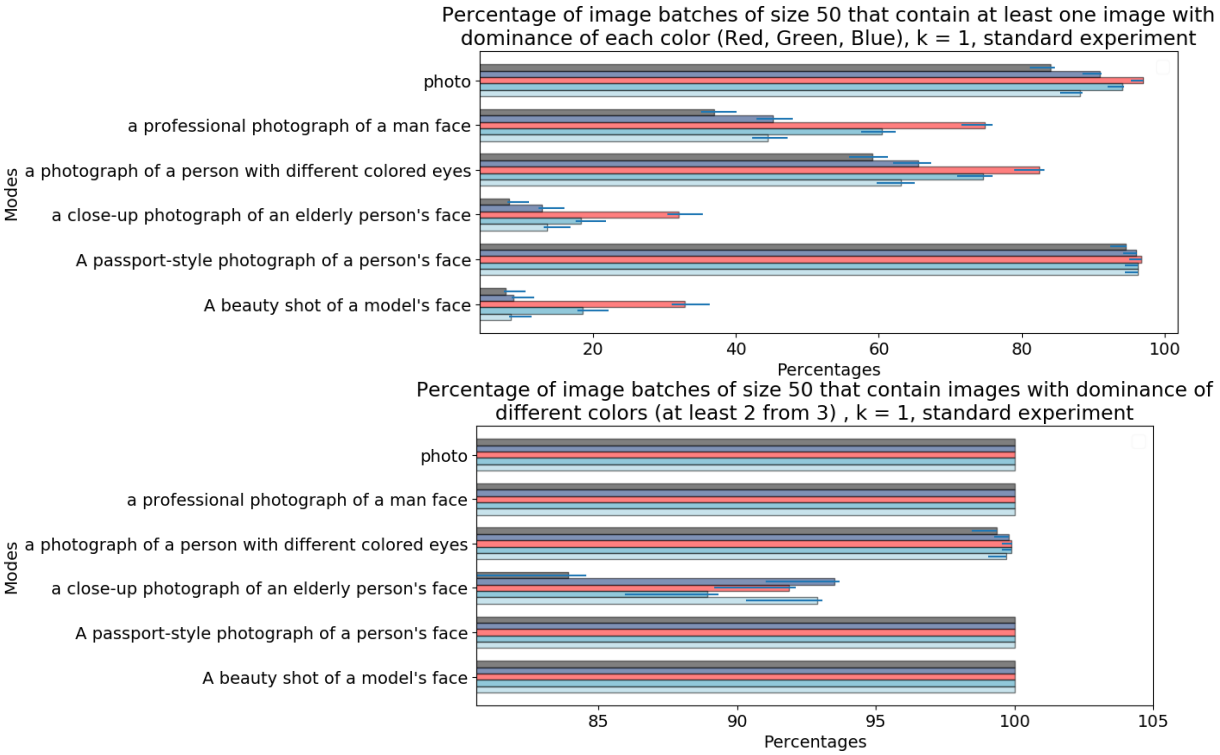

Figure B.7: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1$, batch size $= 50$, standard experiment

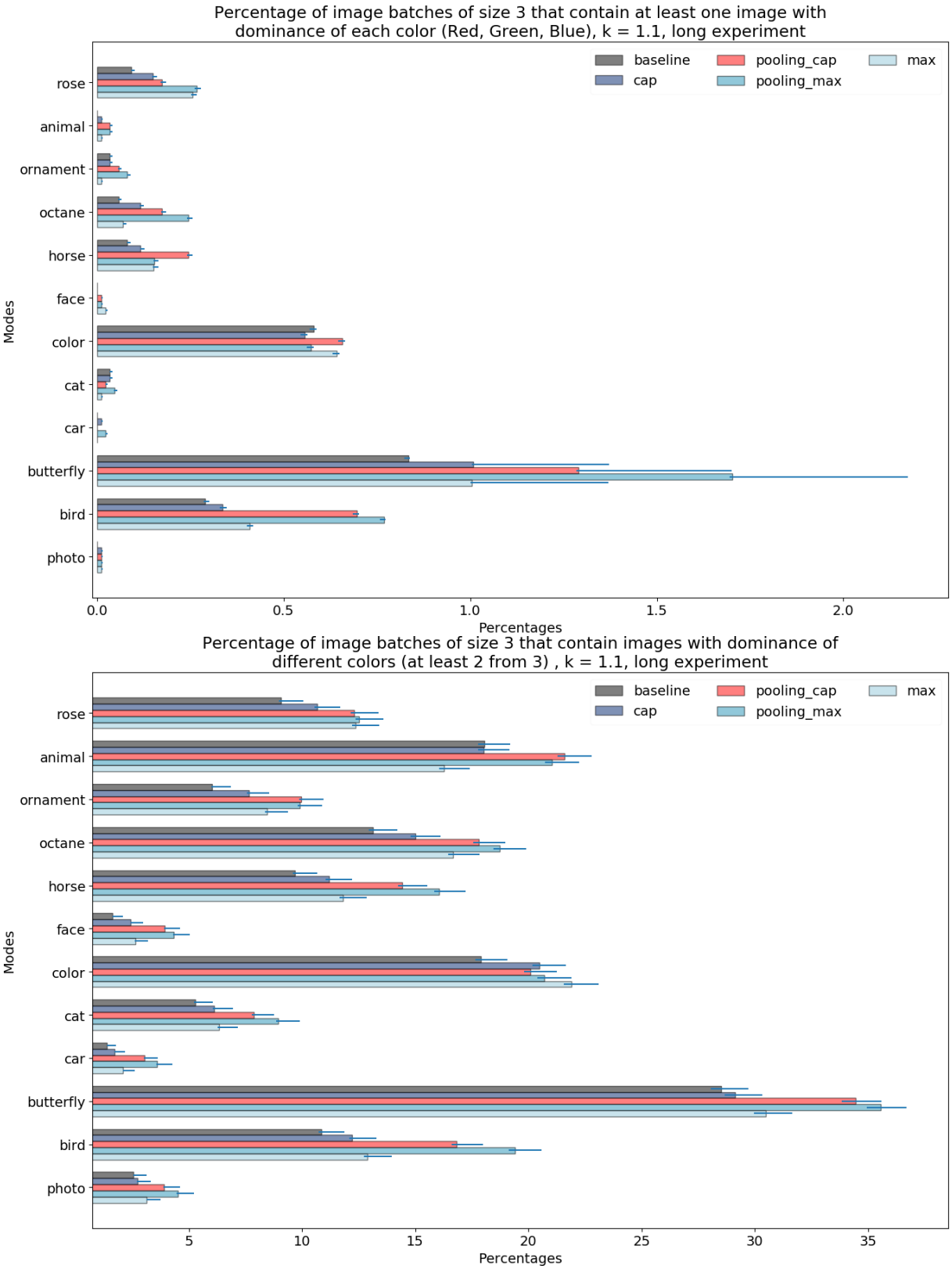

Figure B.8: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 3$, long experiment

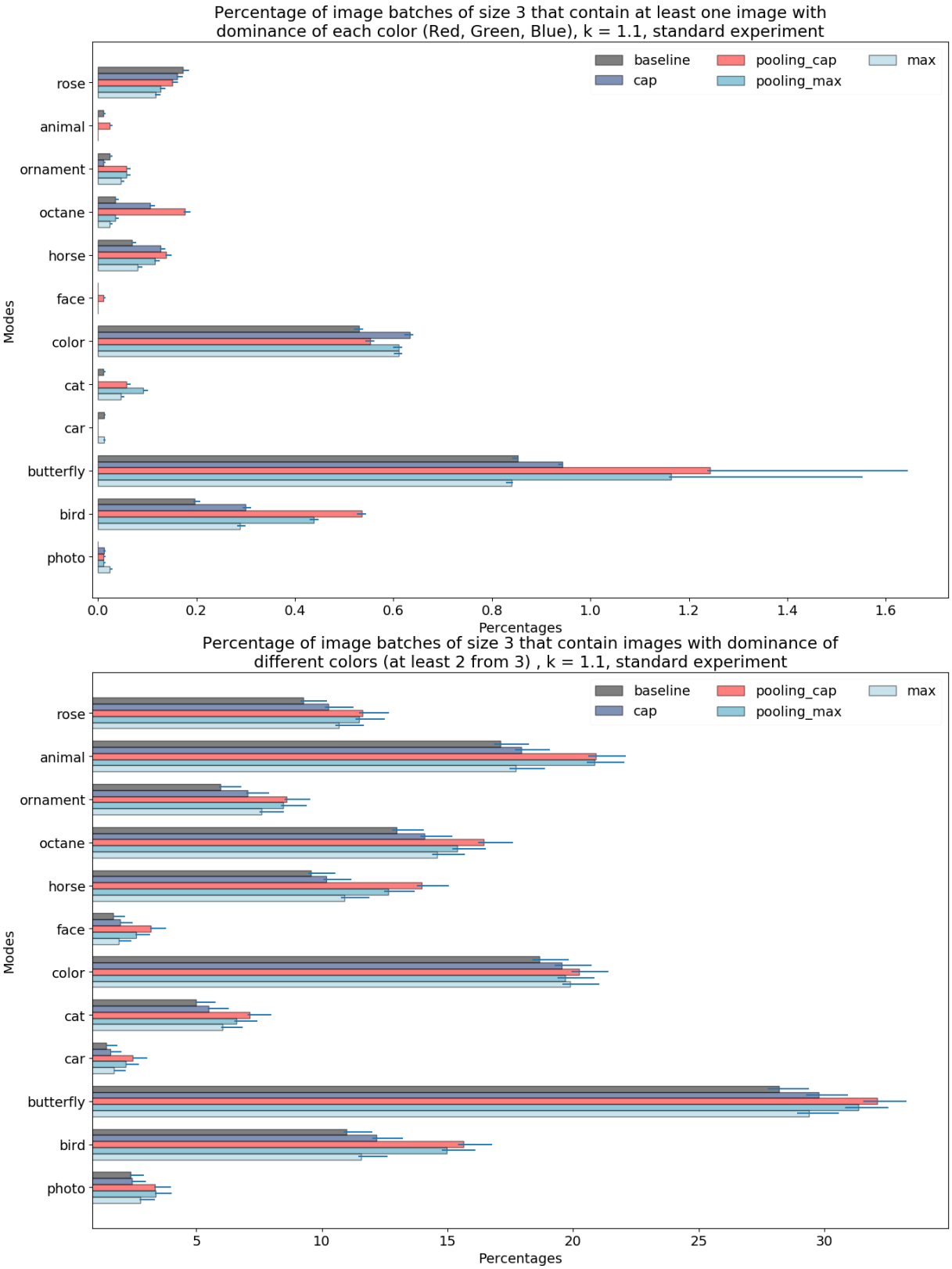

Figure B.9: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 3$, standard experiment

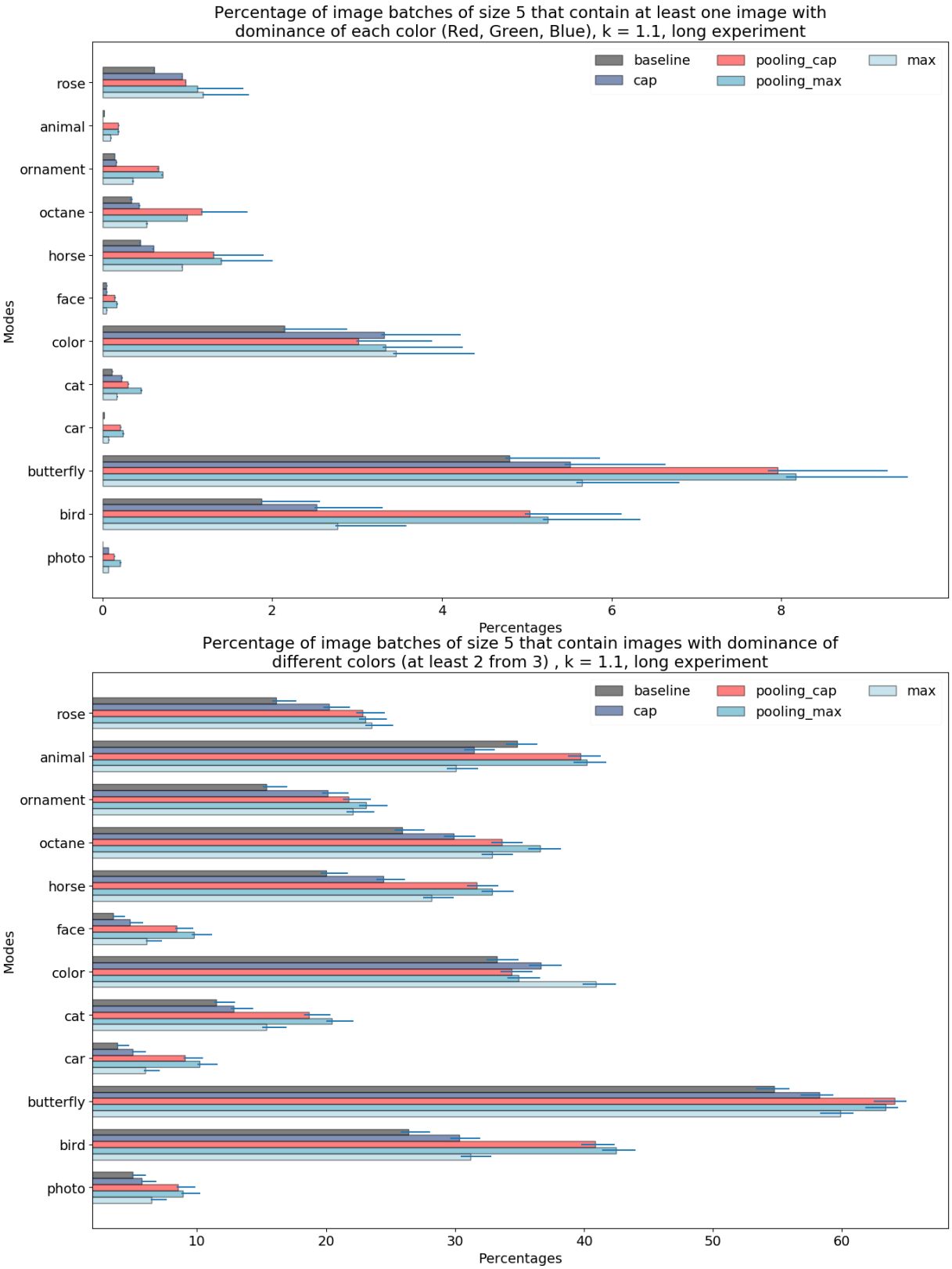

Figure B.10: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 5$, long experiment

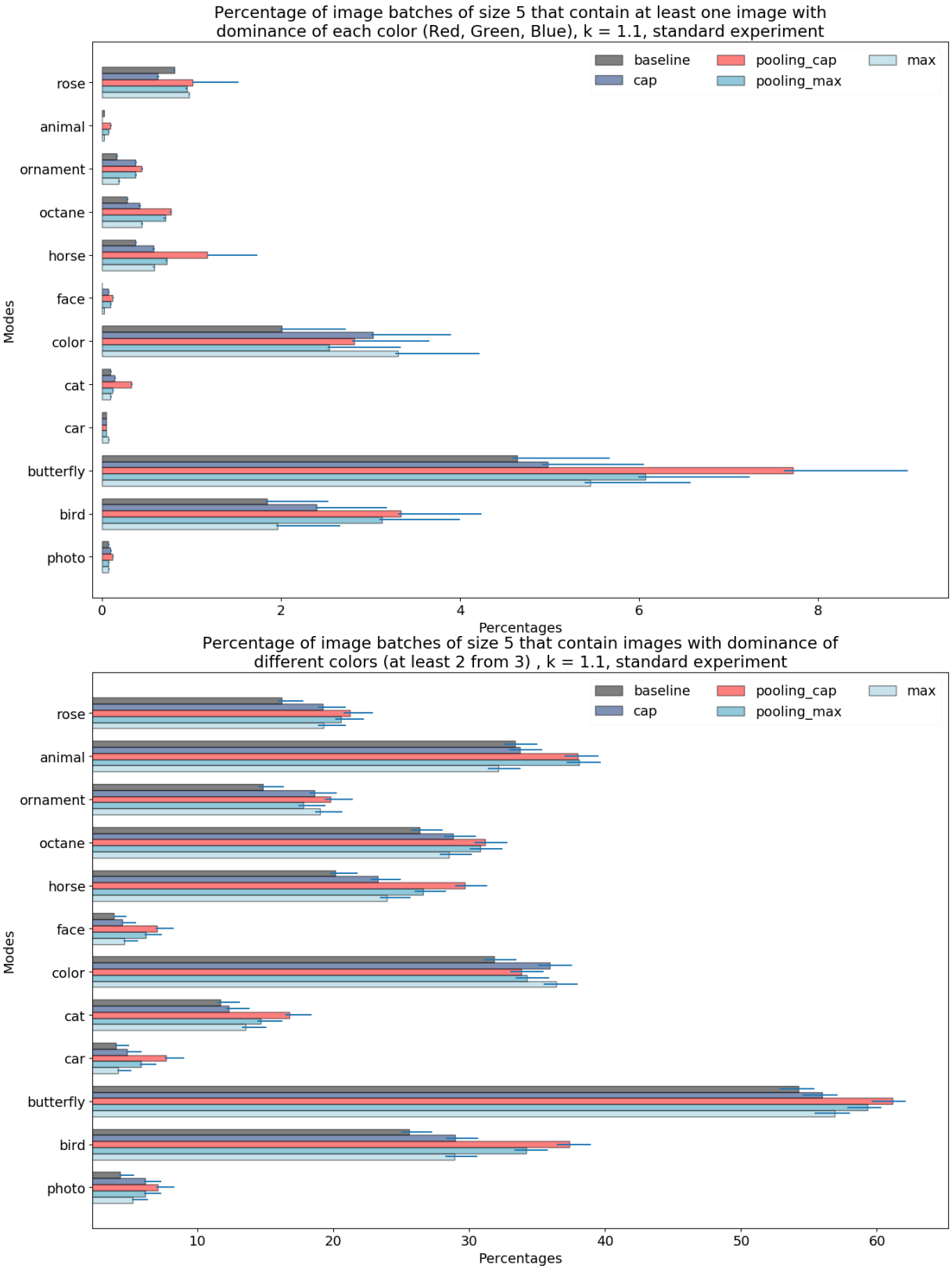

Figure B.11: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 5$, standard experiment

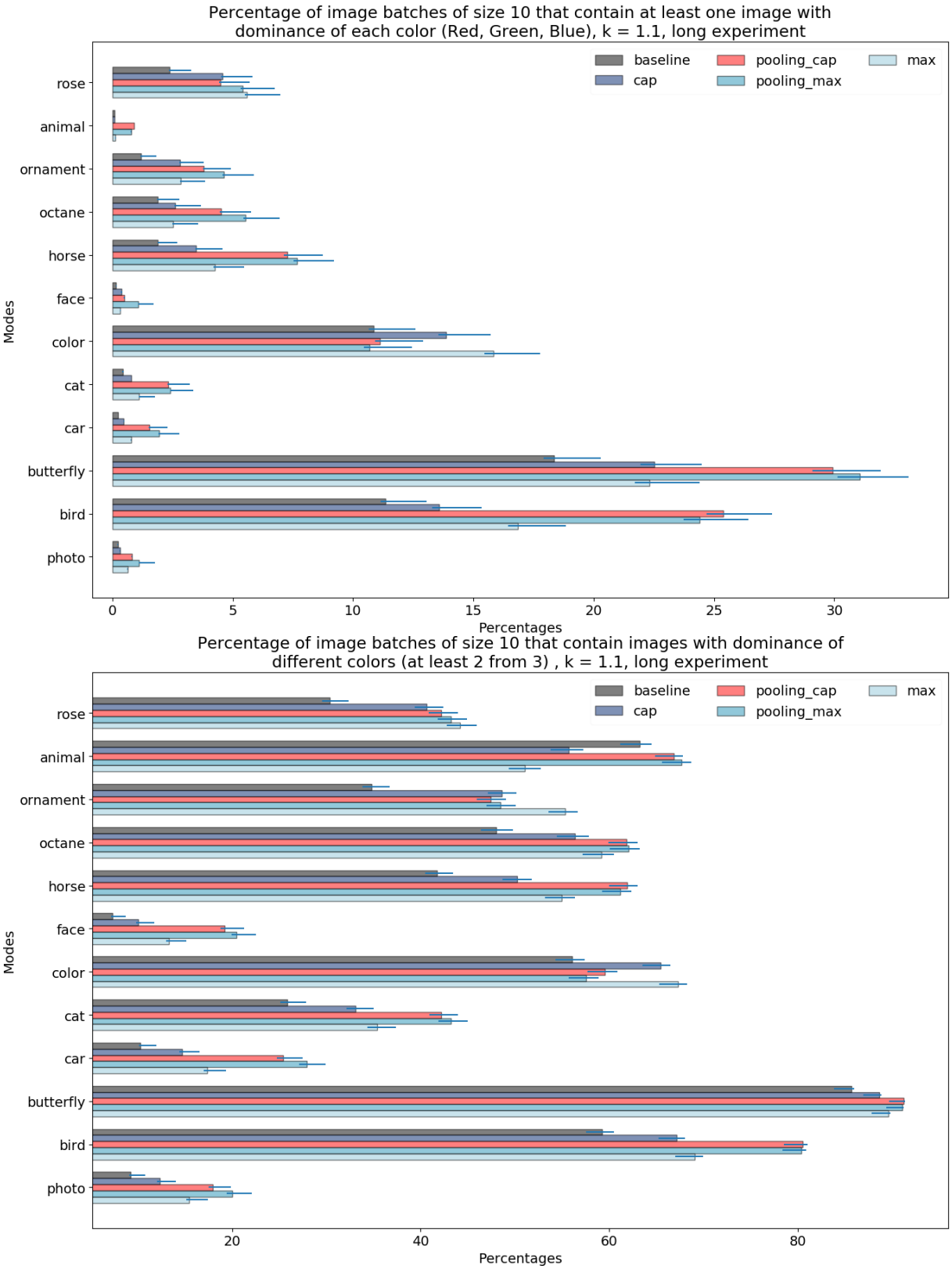

Figure B.12: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 10$, long experiment.

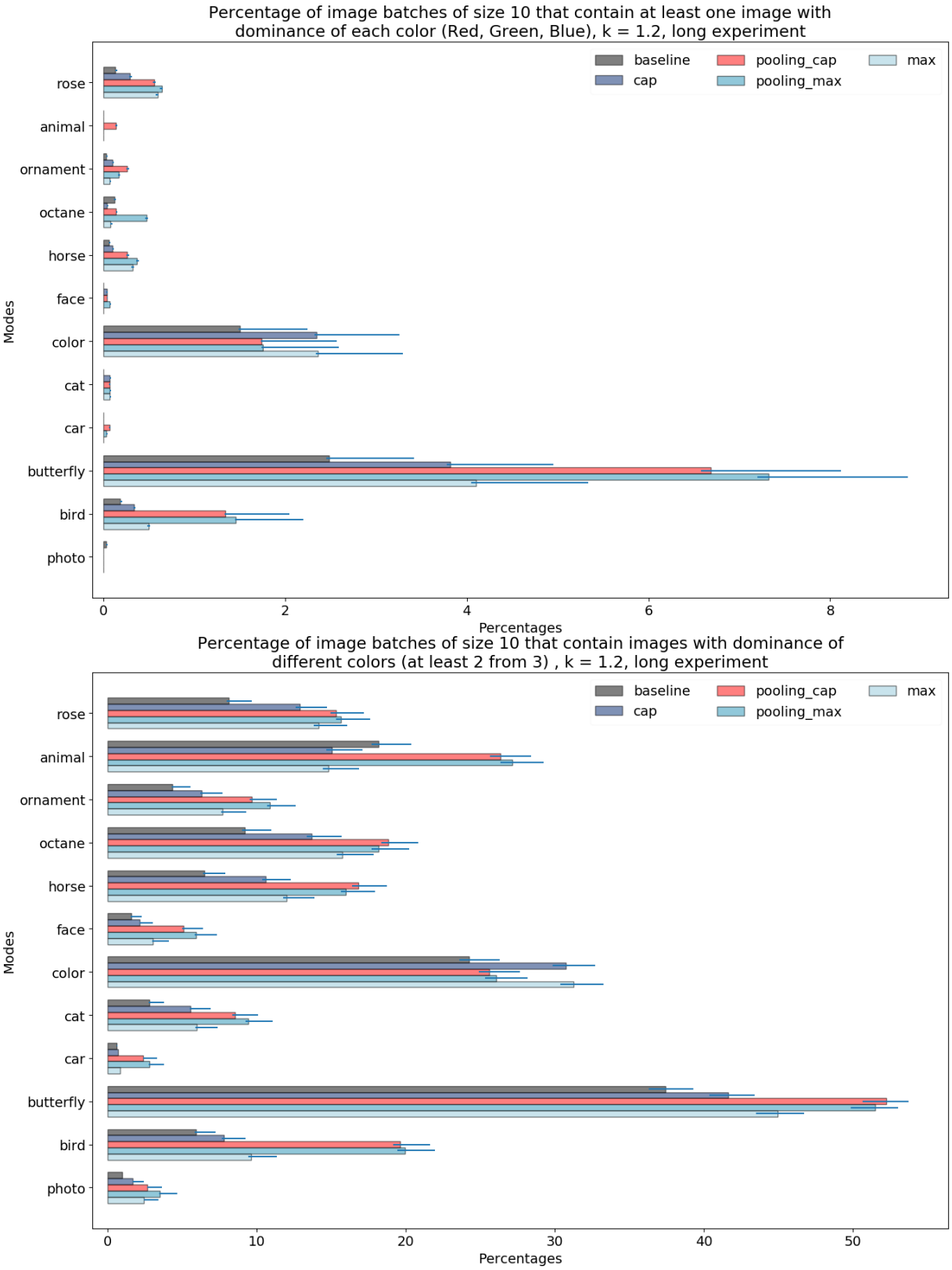

Figure B.13: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size $= 10$, long experiment.

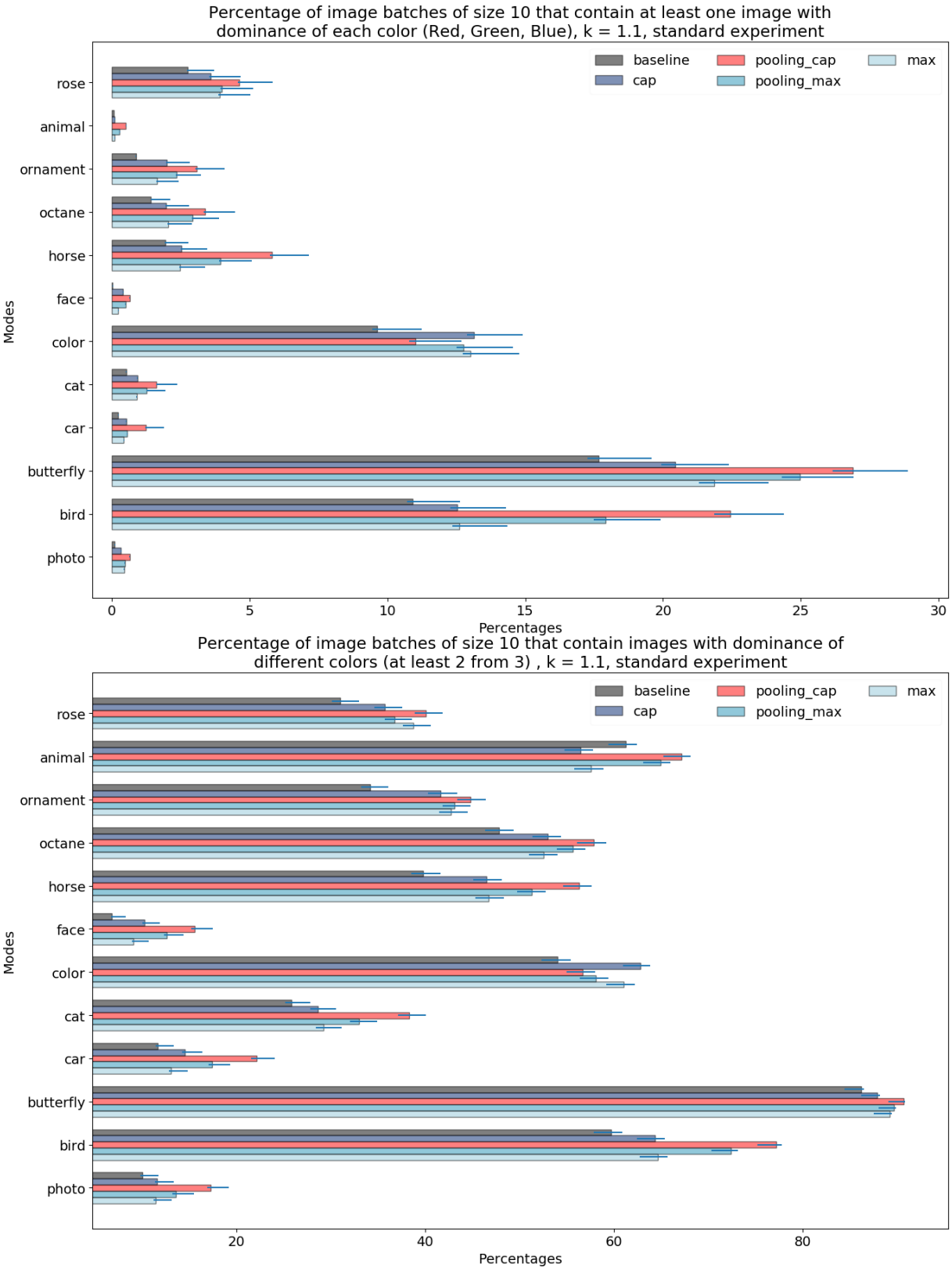

Figure B.14: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size = 10, standard experiment.

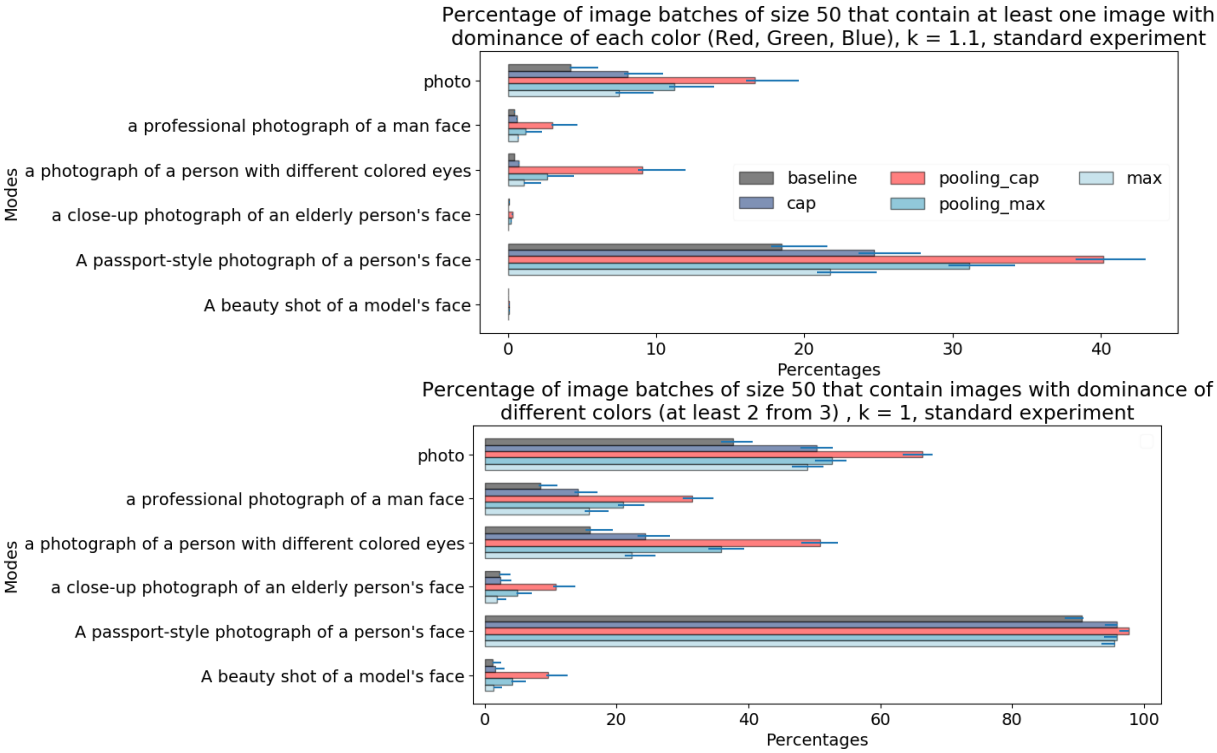

Figure B.15: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.1$, batch size $= 50$, standard experiment.

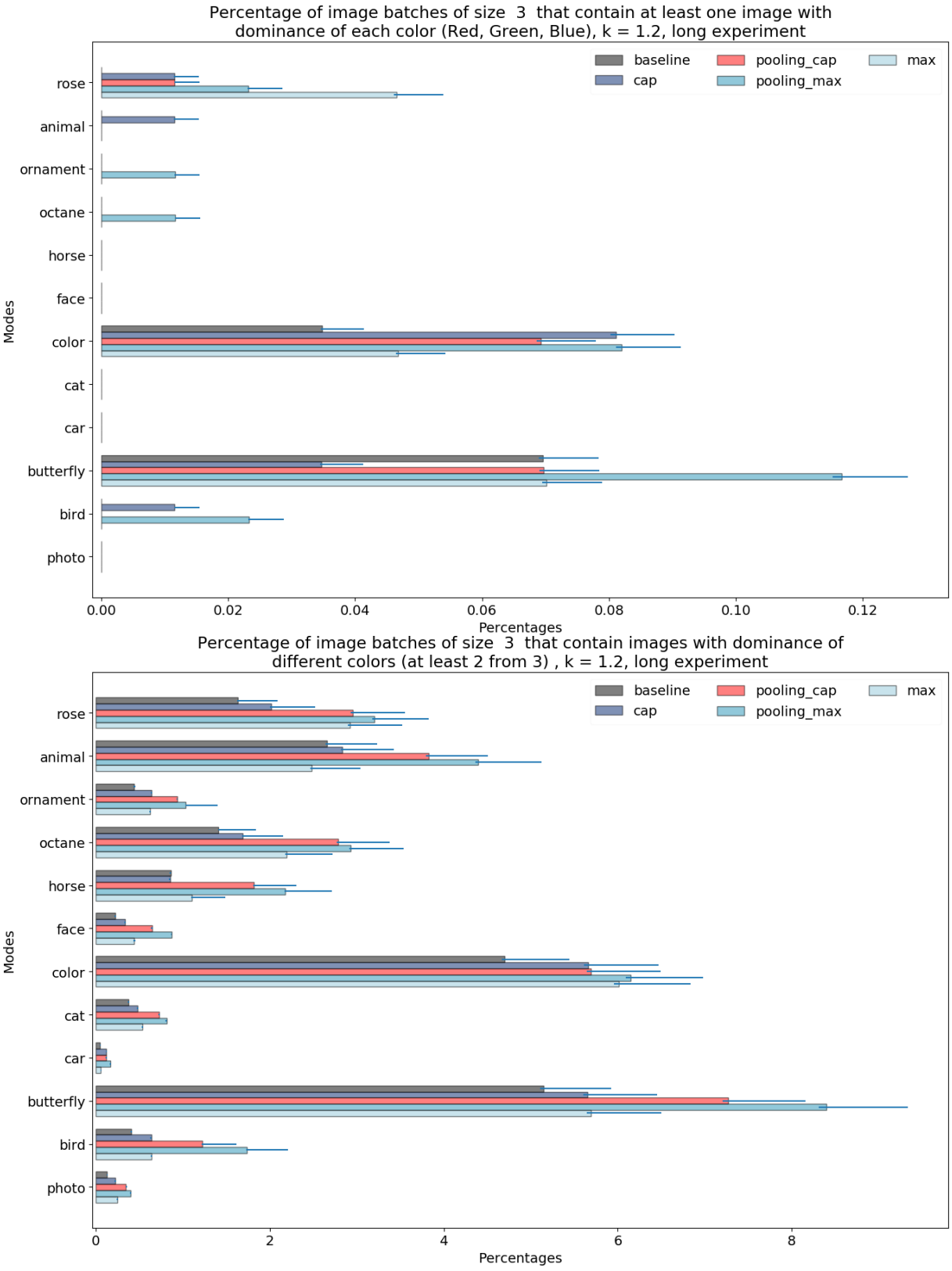

Figure B.16: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size $= 3$, long experiment.

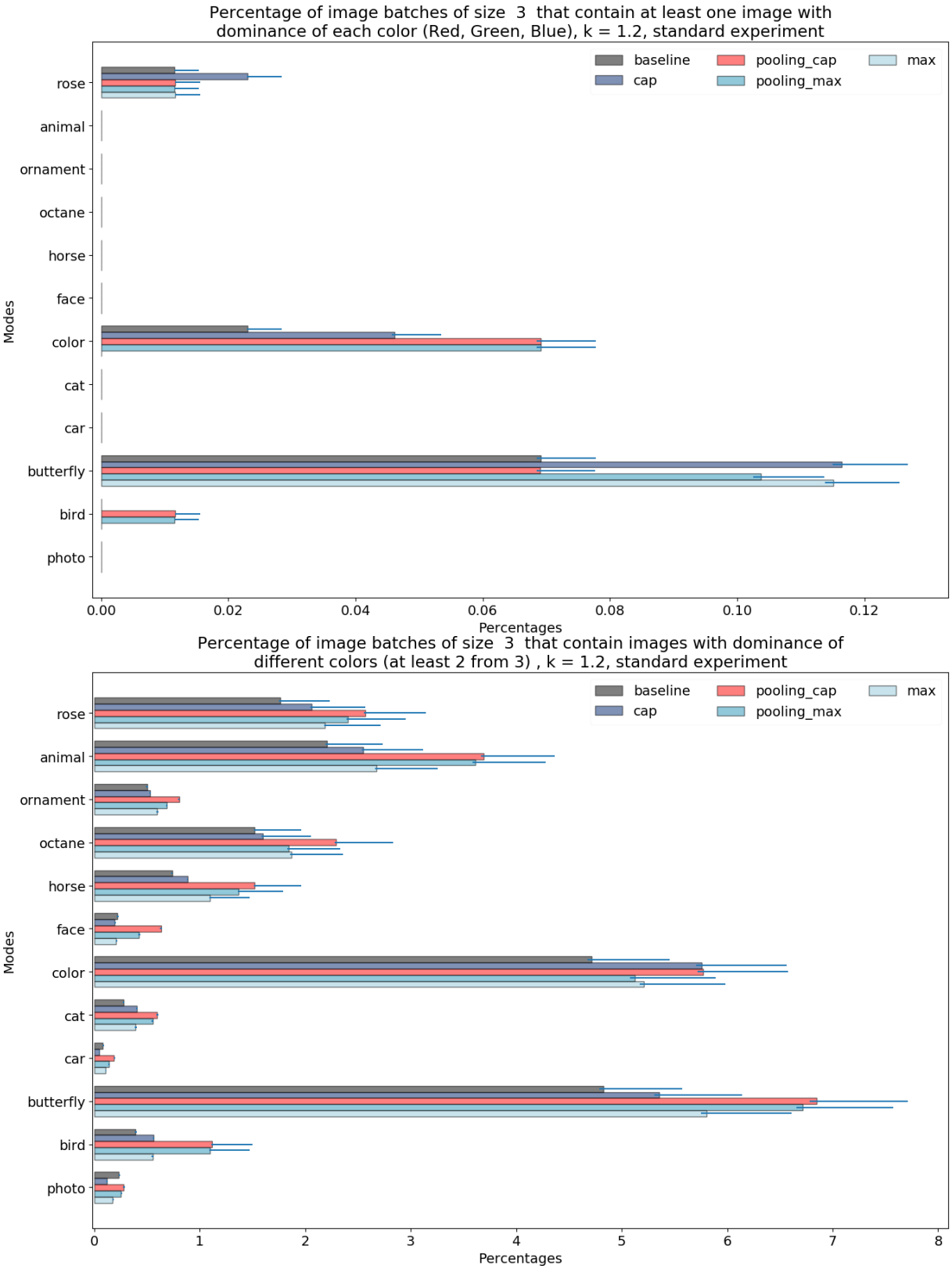

Figure B.17: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size $= 3$, standard experiment.

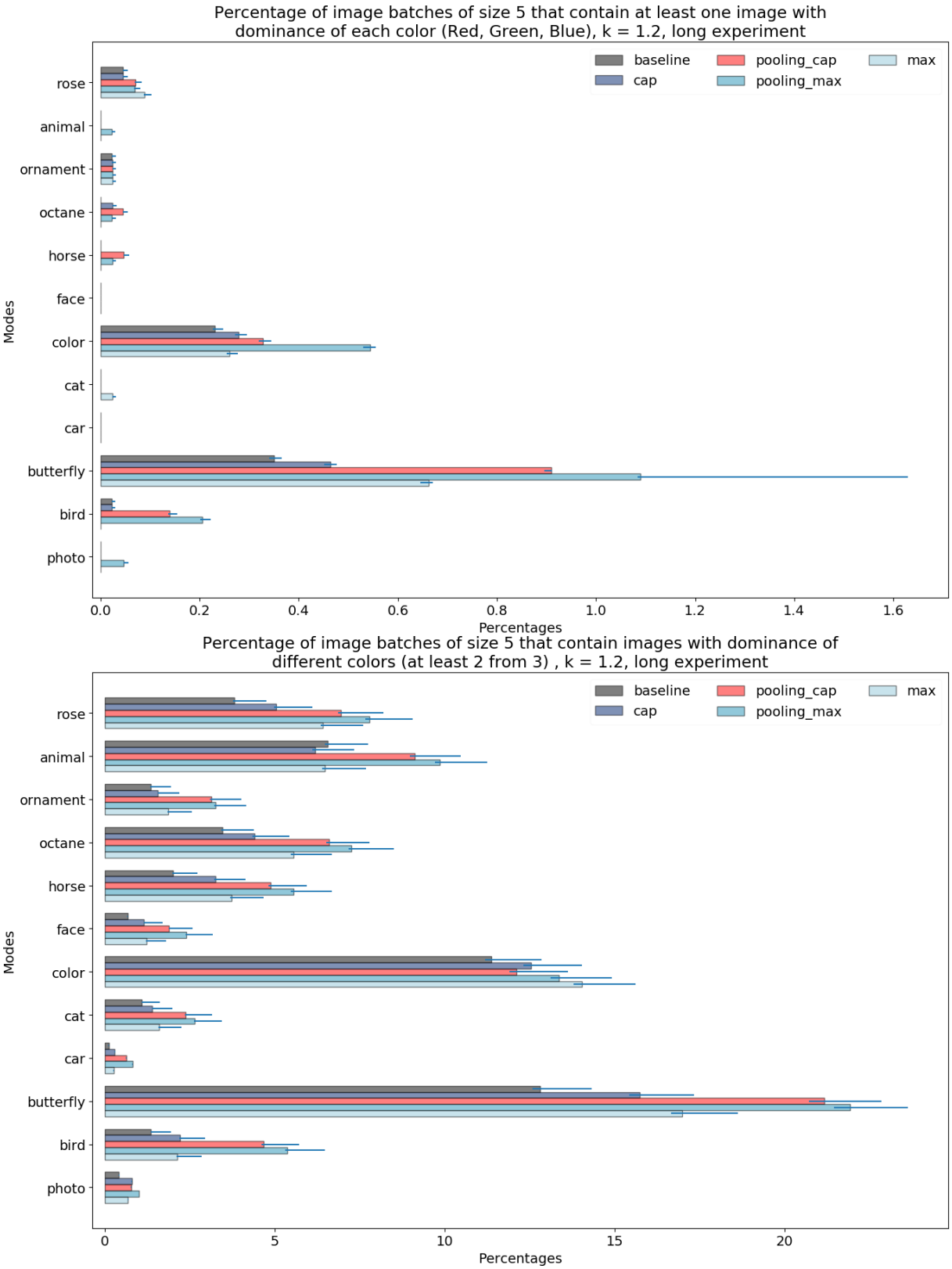

Figure B.18: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size $= 5$, long experiment.

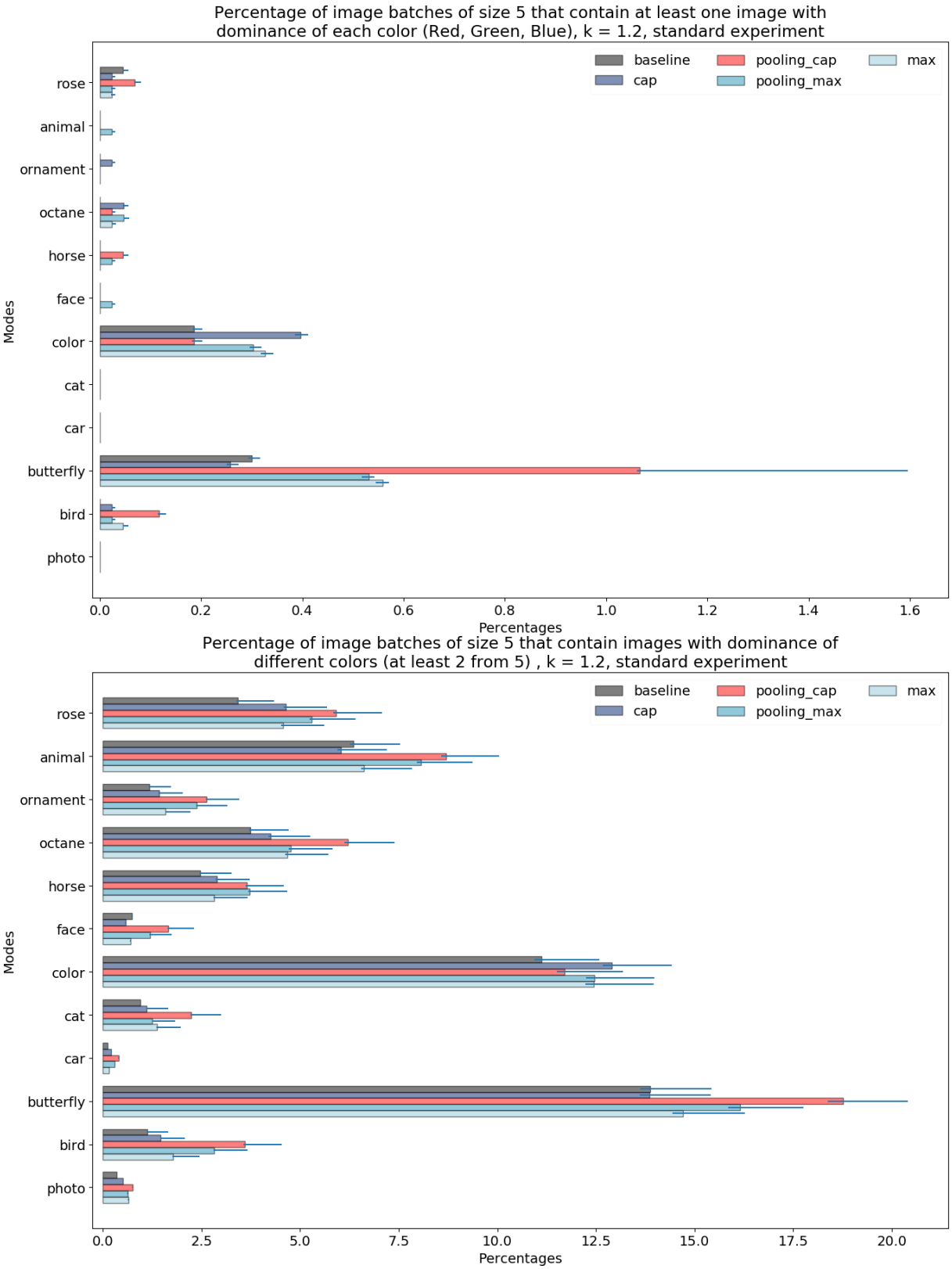

Figure B.19: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size $= 5$, standard experiment.

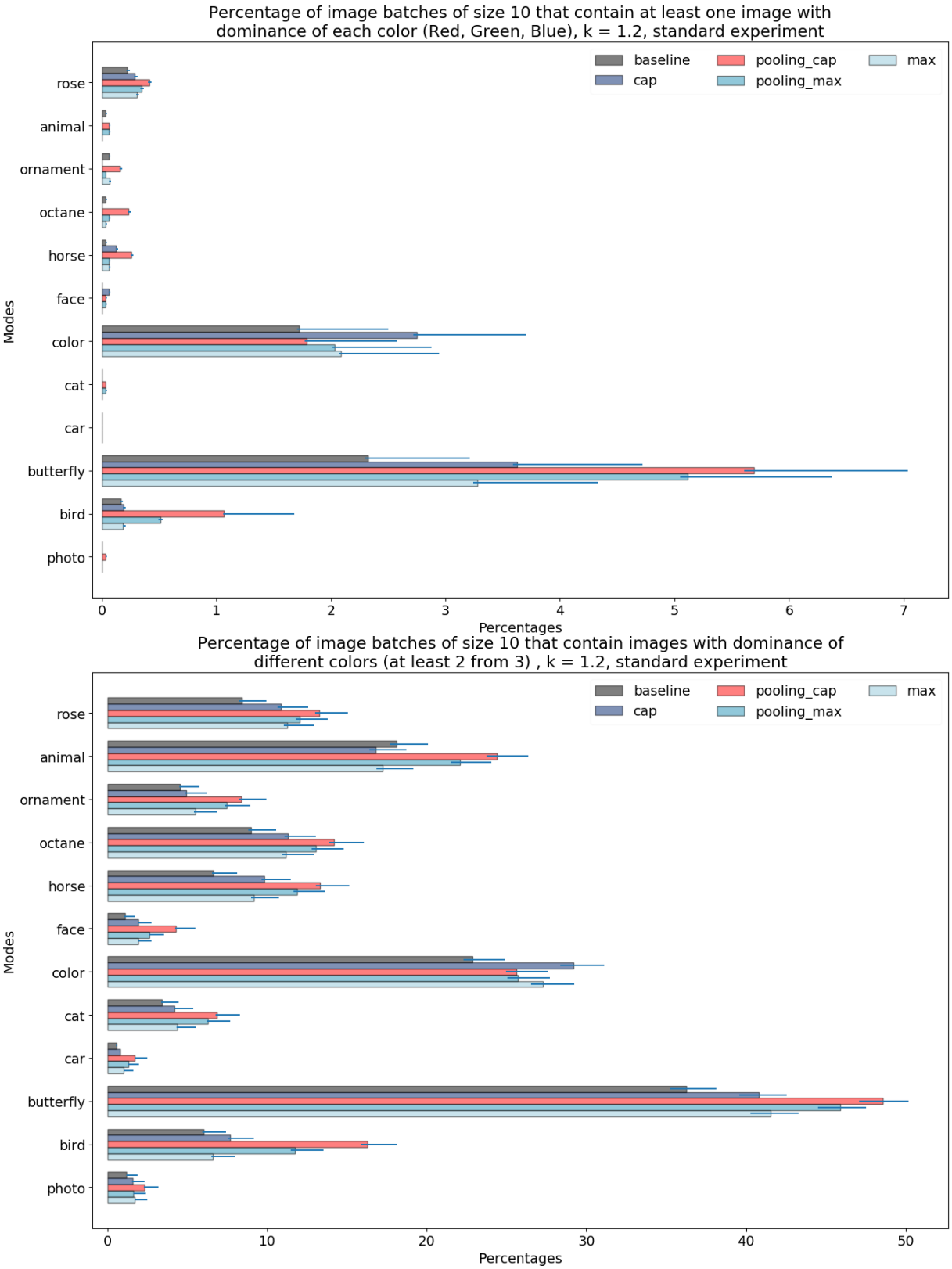

Figure B.20: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size = 10, standard experiment.

Figure B.21: Comparison of different modes for various prompts in regard to the percentage of batches containing images with different dominant colors for the following parameters: $K = 1.2$, batch size = 50, standard experiment.

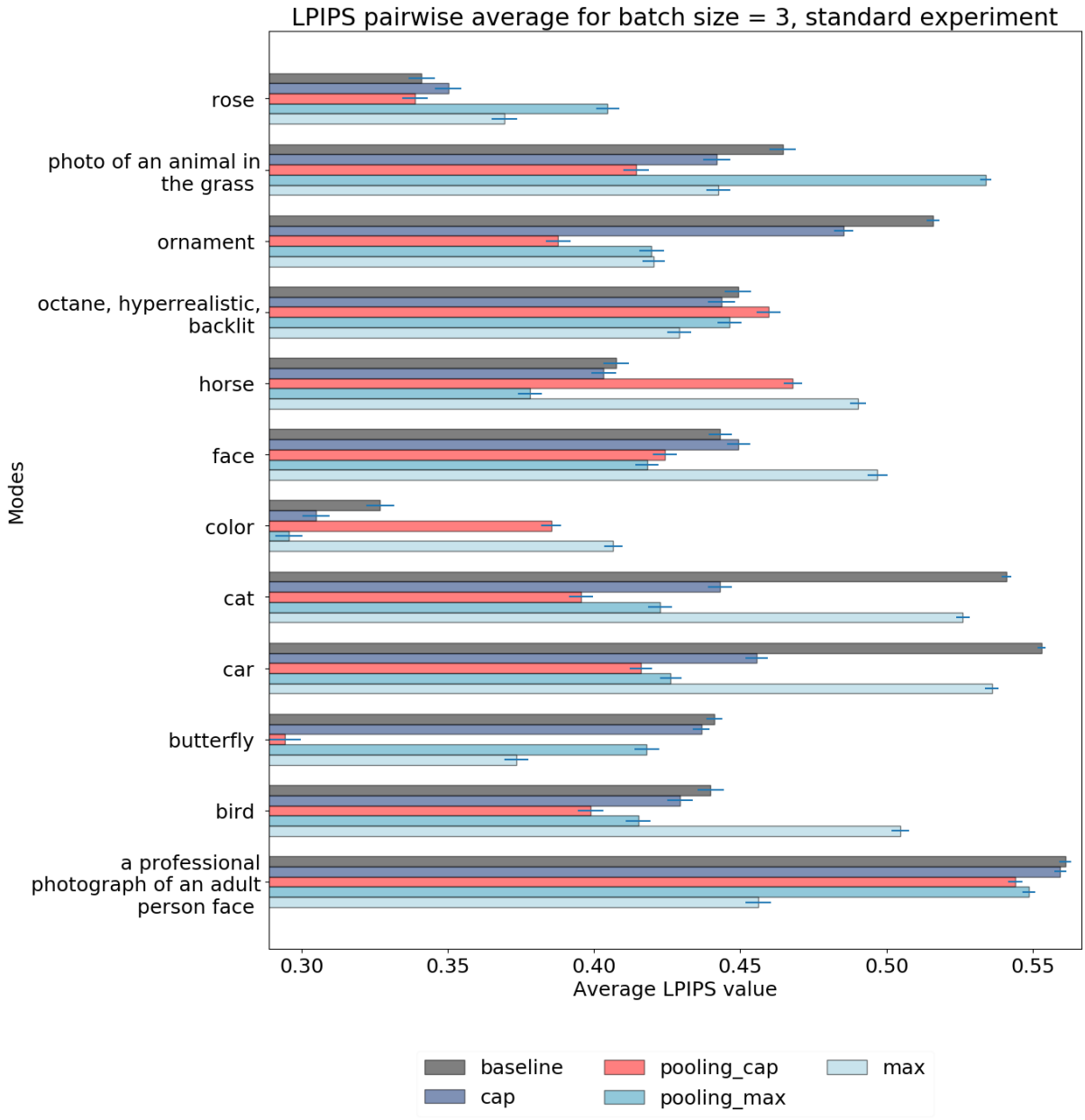

Figure B.22: Average batch pairwise LPIPS, batch size=3, standard experiment : no clear conclusion overall, due to the small batch size.

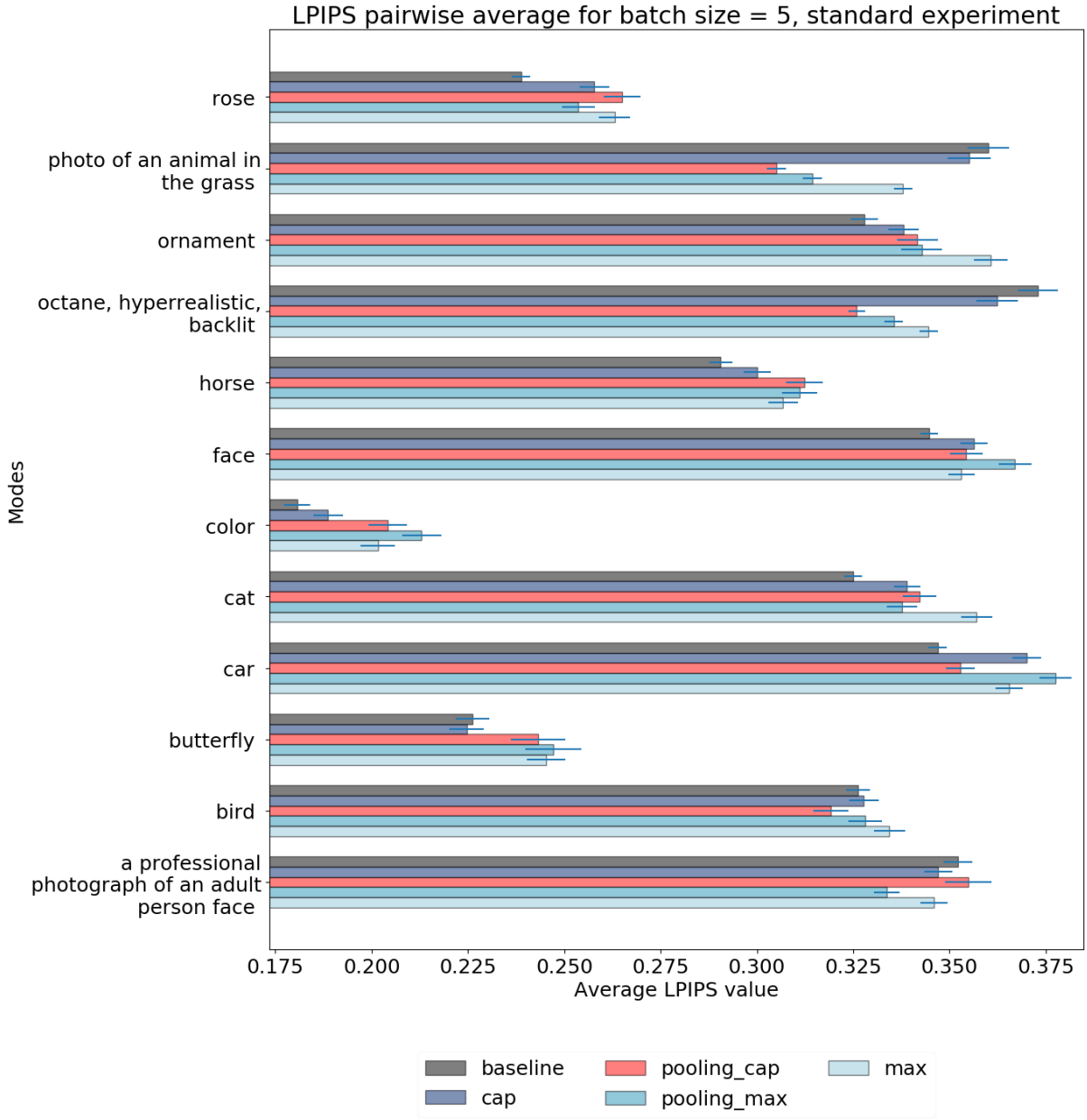

Figure B.23: Average batch pairwise LPIPS, batch size=5, standard experiment

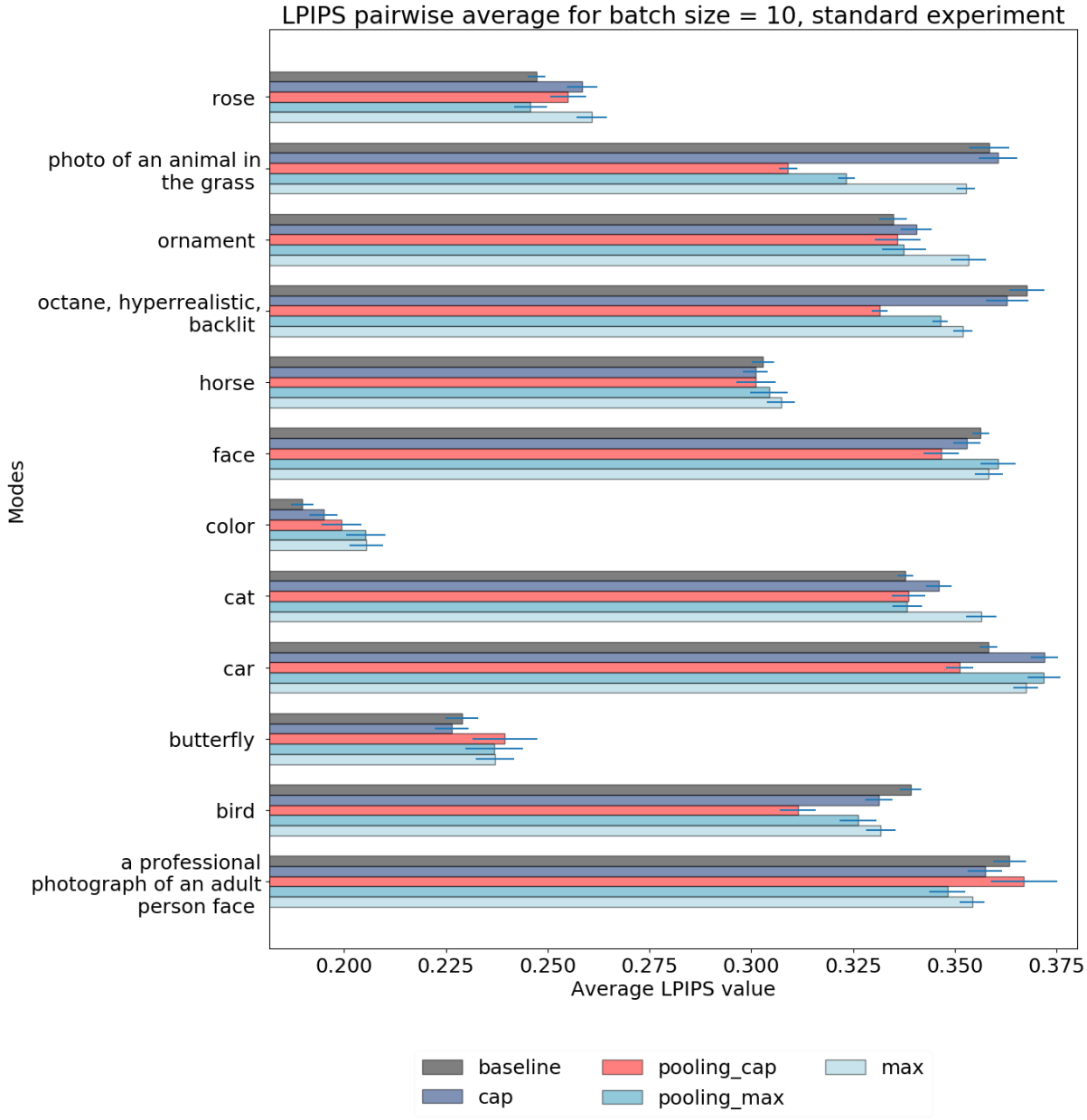

Figure B.24: Average batch pairwise LPIPS, batch size=10

C

Generative Image Privacy

C.1 Overview

Facial image privacy protection is a multi-objective problem combining image quality preservation and privacy robustness against various image recognition systems:

- In Sections C.1.1-C.1.3, we present quantitative results (table of recall/percentage, showing privacy performance).
- Then, in Sections C.1.4.1-C.1.4.3, we present images, showing the image quality.

C.1.1 Experiment 1: additional and extended tables of results

In this subsection, we present transfer results for Experiment 1. All the tables here are similar to the Table 4.2 except for the different choice of transfer embeddings used in recognition.

To be precise, we evaluate the transfer results of PrivacyGAN equipped with generative methods VQGAN and StyleGAN optimised with FaceNet[SKP15] embedding and compare them to the transfer results of Fawkes. The criterion is the transfer to other embeddings than FaceNet, namely ArcFace[DGXZ19], MobileFaceNet[CLGH18], and Resnet_152[HZRS16].

We also present the results of Fawkes combination with generative methods. We note that combining Fawkes poisoning with our methods can be beneficial for facial image privacy protection.

Table C.1 presents results of the transfer to ArcFace embedding; Table C.2 presents results of the transfer to MobileFaceNet embedding; and Table C.3 presents the results of transfer to Resnet_152 embedding method.

From the results of this experiment, we conclude that generative methods optimised with one single embedding do not provide strong privacy protection, but their results are still better than the results of Fawkes.

C.1.2 Experiment 2 (comparing PrivacyGAN equipped with StyleGAN and PrivacyGAN equipped with VQGAN optimised with 2 embedding methods on the LFW dataset): additional tables of results

In the main part of the chapter, we presented the results of Experiment 2 without transfer for embedding method MagFace in Table 4.4 and for embedding method MobileFaceNet in Table 4.5 and with transfer for embedding method SphereFace[LWY+17] in Table 4.6. Here we present other examples with transfer for embedding methods FaceNet, ArcFace, and ResNet_152 in Tables C.4, C.5 and C.6.

From the results of the experiment 2, we conclude that generative methods optimised with two different embeddings provide stronger privacy protection than those optimised with a single embedding method (as in experiment 1).

C.1.3 Experiment 3 (comparing PrivacyGAN, AMT-GAN, and Fawkes on CC dataset): additional tables of results

In this section, we present the results of experiment 3. All the tables here are similar to Table 4.7 of the main part, except for the differences in choice of transfer embeddings.

We present results for generative methods with a criterion based on transfer from MobileFaceNet and MagFace (used in our privacy algorithm) to embeddings FaceNet, ArcFace and ResNet_152 (used in the recognition) in Tables C.7, C.9 and C.11 as well as results without transfer for embeddings MagFace and MobileFaceNet in Tables C.8 and C.10.

Overall, results for the CC dataset are similar to those for LFW, and generative methods remain preferable for privacy protection.

C.1.4 Image examples for original and modified images using both pixel-based and generative methods

The examples in this section show that generative methods modify image features more than the pixel-based methods. Nonetheless, they have less

artificial pixel noise, which is common for images protected by Fawkes. Pixel noise can be more detrimental in terms of visual quality.

C.1.4.1 Modified image examples: experiment 1

In this section, we present original images and their private versions that were obtained in the course of experiment 1 (similarly to Fig. 4.2). In addition, we also provide image examples for combinations of Fawkes and generative methods, namely Fawkes + StyleGAN (F+S), Fawkes + VQ-GAN (F+V), StyleGAN + Fawkes (S+F) and VQGAN + Fawkes (V+F). We see that adding Fawkes on top of generative methods improves image privacy, as in methods StyleGAN + Fawkes (S+F) and VQGAN + Fawkes (V+F). Tthe mentioned image examples can be found in Figs C.1, C.2, C.3 and C.4.

C.1.4.2 Modified image examples: experiment 2

In this section, we present the original images and their private versions that were obtained during experiment 2, in addition to the images that were presented in the main chapter in Fig 4.3. These images can be found in Figs C.5, C.6, C.7 and C.8.

C.1.4.3 Modified image examples: experiment 3

Here, we present more image examples obtained by the procedure described in experiment 3. They are obtained the same way as images in Fig. 4.4. These examples are presented in Figs C.9, C.10 and C.11.

Figure C.1: Experiment 1: Examples of original images from the LFW dataset and their counterparts modified by different privacy methods: StyleGAN_0.003_500, VQGAN_0.005_128, StyleGAN, VQGAN (using FaceNet as an embedding method for optimisation), Fawkes, and combinations of Fawkes with generative methods. Here we can see that while generative methods in general add more modification to an image than Fawkes, generative methods produce realistic images and do not add pixel noise.

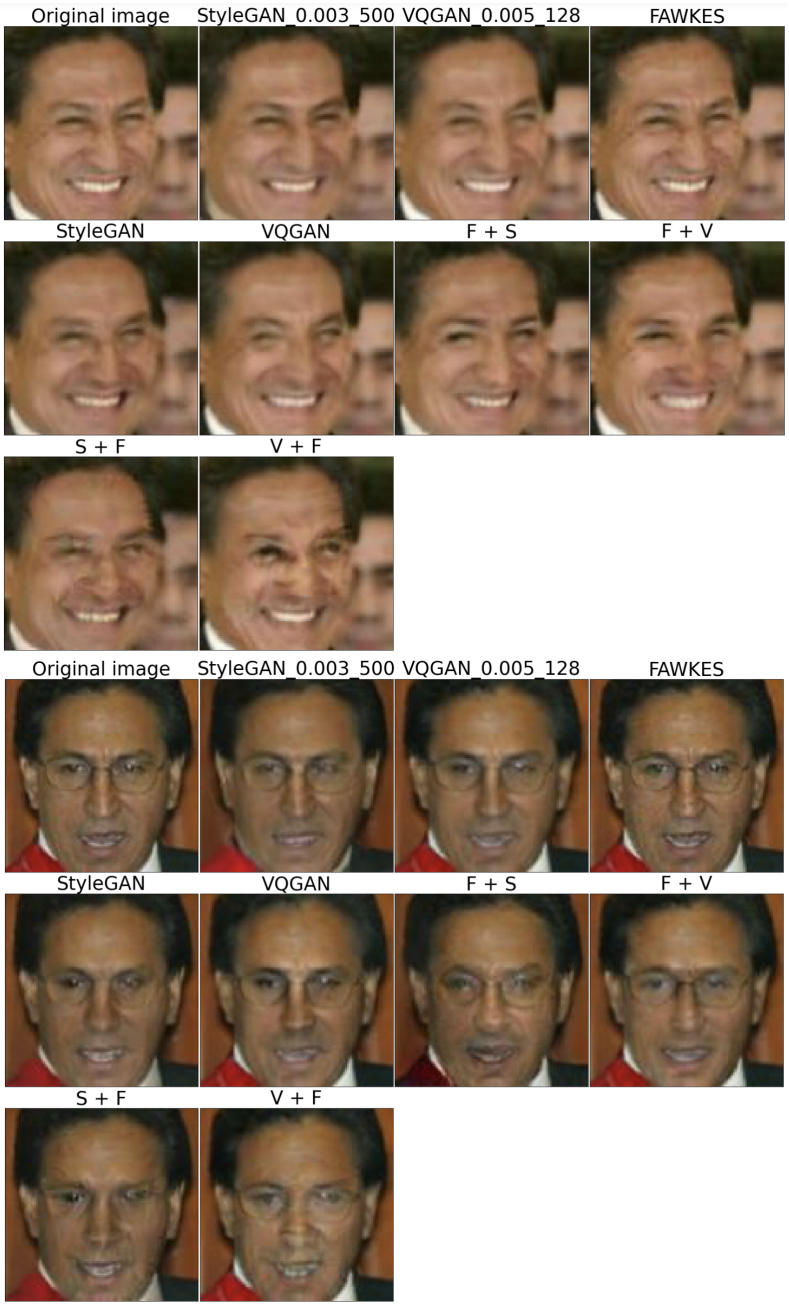

Figure C.2: Experiment 1: Examples of original images from the LFW dataset and their counterparts modified by different privacy methods: StyleGAN_0.003_500, VQGAN_0.005_128, StyleGAN, VQGAN (using FaceNet as an embedding method for optimisation), Fawkes, and combinations of Fawkes with generative methods. Here we can see that while generative methods in general add more modification to an image than Fawkes, generative methods produce realistic images and do not add pixel noise.

Figure C.3: Experiment 1: Examples of original images from the LFW dataset and their counterparts modified by different privacy methods: StyleGAN_0.003_500, VQGAN_0.005_128, StyleGAN, VQGAN (using FaceNet as an embedding method for optimisation), Fawkes, and combinations of Fawkes with generative methods. Here we can see that while generative methods in general add more modification to an image than Fawkes, generative methods produce realistic images and do not add pixel noise.

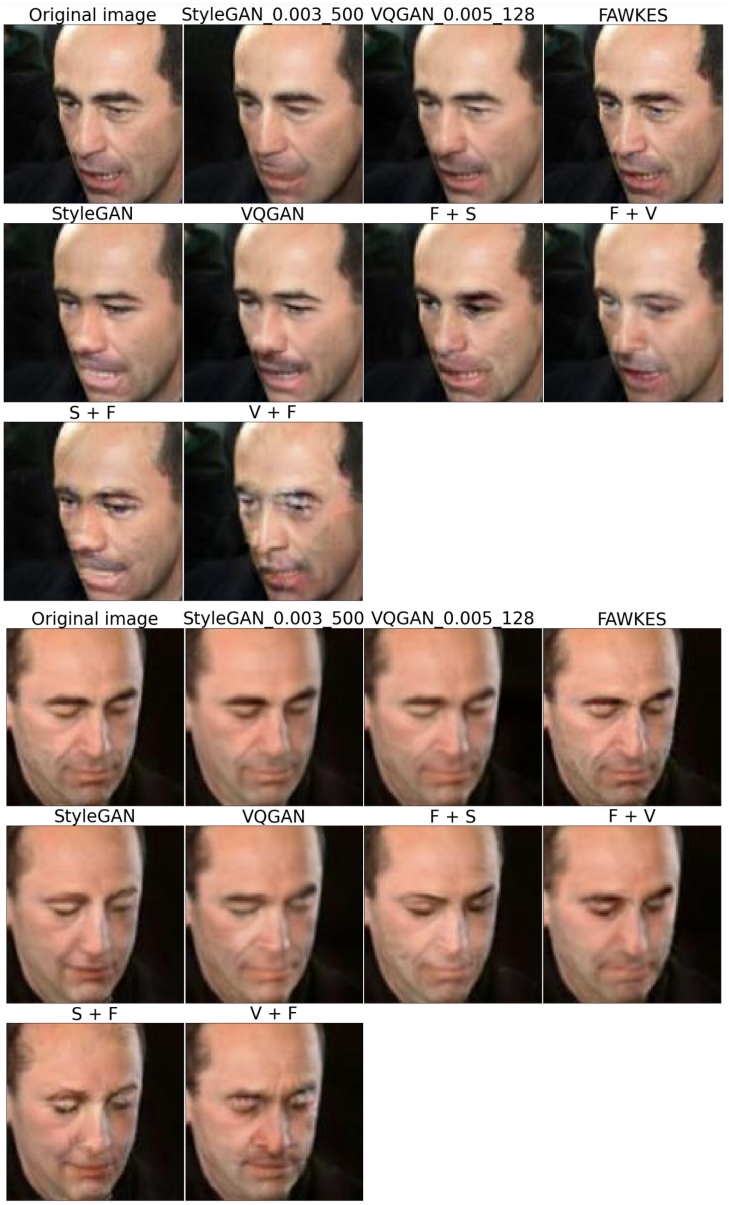

Figure C.4: Experiment 1: Examples of original images from the LFW dataset and their counterparts modified by different privacy methods: StyleGAN_0.003_500, VQGAN_0.005_128, StyleGAN, VQGAN (using FaceNet as an embedding method for optimisation), Fawkes, and combinations of Fawkes with generative methods. Here we can see that while generative methods in general add more modification to an image than Fawkes, generative methods produce realistic images and do not add pixel noise.

Figure C.5: Experiment 2: Examples of images from the LFW dataset: the original image and the image modified by different privacy methods: StyleGAN, StyleGAN_0.02_500, StyleGAN_0.02_1000, VQGAN, VQGAN_0.03_512, VQGAN_0.04_128 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection.

Figure C.6: Experiment 2: Examples of images from the LFW dataset: the original image and the image modified by different privacy methods: StyleGAN, StyleGAN_0.02_500, StyleGAN_0.02_1000, VQGAN, VQGAN_0.03_512, VQGAN_0.04_128 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection.

Figure C.7: Experiment 2: Examples of images from the LFW dataset: the original image and the image modified by different privacy methods: StyleGAN, StyleGAN_0.02_500, StyleGAN_0.02_1000, VQGAN, VQGAN_0.03_512, VQGAN_0.04_128 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection.

Figure C.8: Experiment 2: Examples of images from the LFW dataset: the original image and the image modified by different privacy methods: StyleGAN, StyleGAN_0.02_500, StyleGAN_0.02_1000, VQGAN, VQGAN_0.03_512, VQGAN_0.04_128 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection.

Figure C.9: Experiment 3: Examples of images of volunteers modified by various privacy methods, including AMT-GAN, Fawkes, StyleGAN_0.02_1000, VQGAN_0.003_128, VQGAN_0.04_4096 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection. We can also see that some images are modified more than others after applying privacy-protection methods.

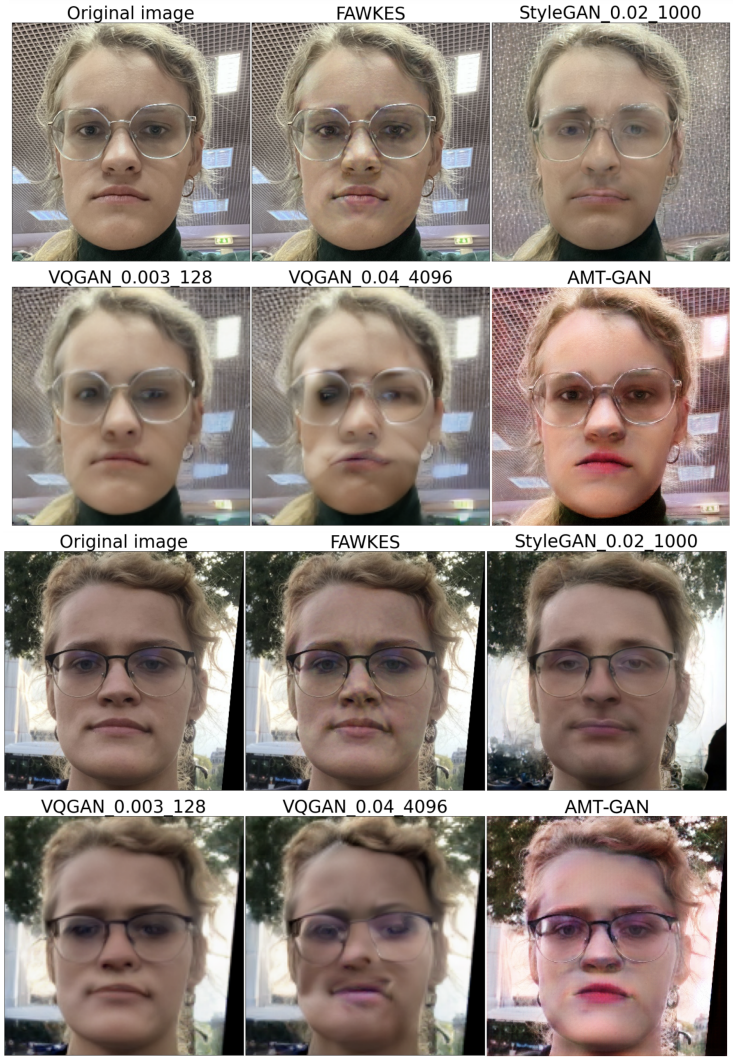

Figure C.10: Experiment 3: Examples of images of volunteers modified by various privacy methods, including AMT-GAN, Fawkes, StyleGAN_0.02_1000, VQGAN_0.003_128, VQGAN_0.04_4096 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection. We can also see that some images are modified more than others after applying privacy-protection methods.

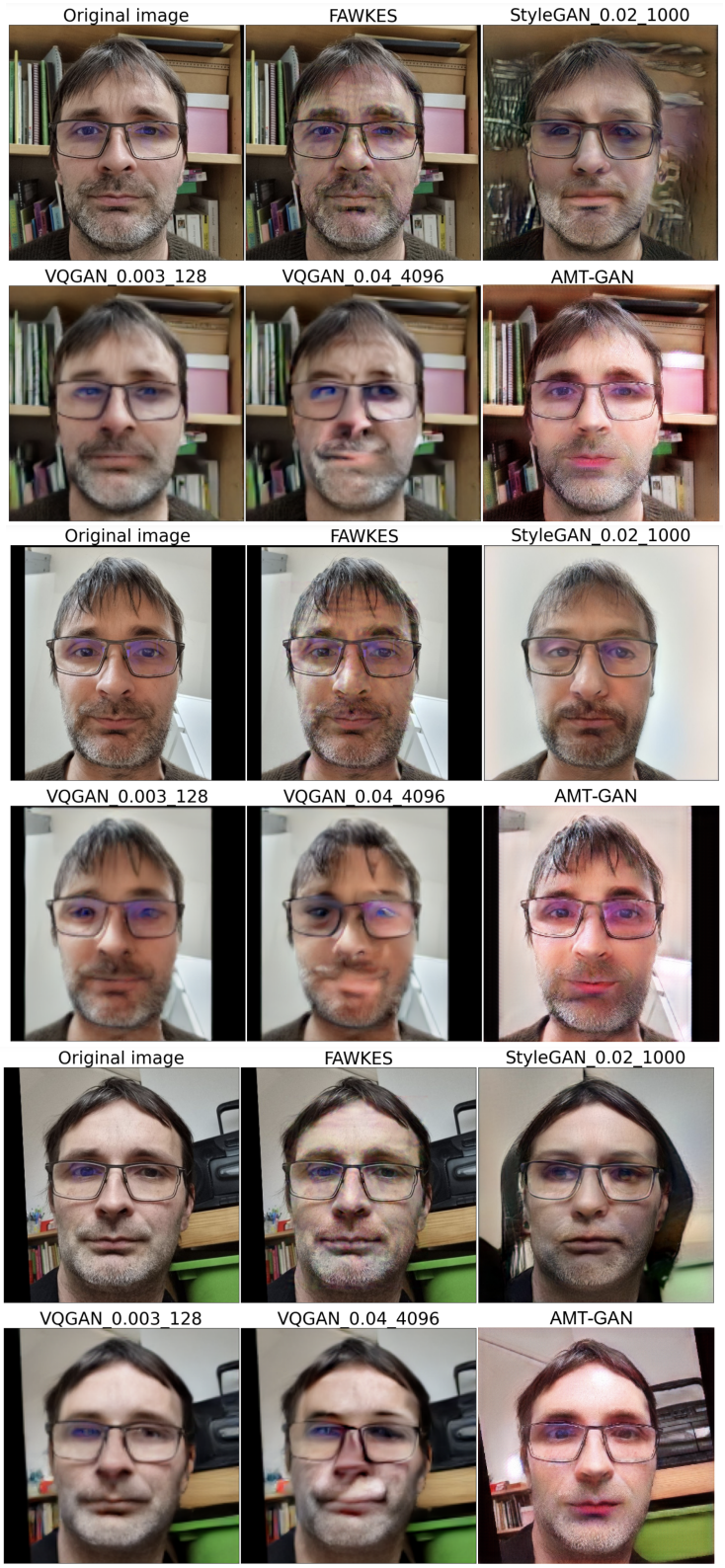

Figure C.11: Experiment 3: Examples of images of volunteers modified by various privacy methods, including AMT-GAN, Fawkes, StyleGAN_0.02_1000, VQGAN_0.003_128, VQGAN_0.04_4096 with embedding methods MagFace and MobileFaceNet. Here, we can see that in some cases, increasing the number of iterations in optimisation and modifying the coefficient K of the embedding method can affect the quality of an image while improving its privacy protection. We can also see that some images are modified more than others after applying privacy-protection methods.

	PrivacyGAN				Pixel-based	Combinations			
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes	F + S	F + V	S + F	V + F
Percentage	5.052	0.917	4.931	0.984	0.936	7.612	7.855	**12.456**	10.397
Recall@1: m.i.	7.962	23.981	8.531	22.464	23.602	5.782	5.118	**2.464**	3.791
Recall@1: o.i.	8.246	24.976	10.095	24.692	24.313	6.445	7.062	**2.749**	4.408
Recall@3: m.i.	18.152	56.019	18.720	57.062	59.953	12.749	11.848	**7.346**	10.095
Recall@3: o.i.	17.536	56.730	19.858	58.578	59.431	11.991	13.175	**6.019**	8.768
Recall@5: m.i.	24.076	69.905	24.265	72.986	76.209	16.919	16.114	**10**	13.081
Recall@5: o.i.	22.038	70.379	24.929	72.559	75.877	15.782	16.445	**8.199**	11.611
Recall@10: m.i.	32.227	79.052	33.365	79.953	83.175	23.649	23.791	**13.886**	19.194
Recall@10: o.i.	30.190	78.389	33.318	79.905	83.033	21.754	22.607	**12.038**	16.635
Recall@50: m.i.	53.791	90.379	55.829	90.995	92.464	42.559	43.223	**28.436**	34.882
Recall@50: o.i.	52.701	91.185	55.924	91.469	92.227	41.706	42.607	**26.398**	31.848
Recall@100: m.i.	63.934	93.602	64.882	93.697	94.929	51.991	52.938	**37.488**	44.313
Recall@100: o.i.	63.318	94.076	65.877	94.123	94.692	51.754	53.602	**35.782**	42.607

Table C.1: Evaluation on LFW dataset in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with FaceNet, and all tests are performed with ArcFace. Lower recall and a higher percentage mean better privacy. As shown in Table C.6 methods that are optimised for two embeddings have a better transfer recall. We see that adding Fawkes on top of generative methods improves image privacy, as in the methods StyleGAN + Fawkes (S+F) and VQGAN + Fawkes (V+F).

	PrivacyGAN				Pixel-based	Combinations			
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes	F + S	F + V	S + F	V + F
Percentage	1.226	0.446	0.947	0.454	0.454	1.875	4.359	**8.645**	5.586
Recall@1: m.i.	19.479	26.872	21.991	26.588	25.592	17.773	**1.185**	7.062	10.332
Recall@1: o.i.	19.100	26.303	21.943	26.493	26.351	19.147	12.938	**5.071**	8.910
Recall@3: m.i.	55.166	72.275	58.815	73.981	72.986	43.033	18.436	**16.825**	23.555
Recall@3: o.i.	51.090	74.692	56.682	72.701	73.697	41.517	26.161	**11.232**	19.005
Recall@5: m.i.	70.616	94.360	74.265	96.209	94.408	55.261	30.853	**22.275**	30.444
Recall@5: o.i.	64.882	93.934	71.896	95.498	93.934	51.896	31.801	**14.929**	24.550
Recall@10: m.i.	77.678	96.303	82.227	97.536	96.351	65.118	42.227	**29.052**	38.341
Recall@10: o.i.	75.024	96.066	80.379	97.156	96.019	61.848	41.185	**21.801**	33.033
Recall@50: m.i.	88.626	98.436	91.517	98.626	97.962	79.763	62.370	**48.483**	58.104
Recall@50: o.i.	87.867	98.389	90.284	98.389	97.725	79.242	61.943	**41.280**	53.460
Recall@100: m.i.	91.659	98.815	94.313	98.815	98.436	84.929	71.090	**55.735**	66.446
Recall@100: o.i.	91.754	98.673	93.223	98.768	98.152	84.408	69.716	**51.043**	62.844

Table C.2: Evaluation on LFW dataset in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with FaceNet, and recognition (for all methods) is tested with MobileFaceNet. Lower recall and a higher percentage mean better privacy. Generative methods do obtain better results than Fawkes. We can also see that adding Fawkes on top of generative methods improves image privacy, as in the methods StyleGAN + Fawkes (S+F) and VQGAN + Fawkes (V+F).

	PrivacyGAN				Pixel-based	Combinations			
	StyleGAN	StyleGAN _0.003_500	VQGAN	VQGAN _0.005_128	Fawkes	F + S	F + V	S + F	V + F
Percentage	0.670	0.342	0.564	0.395	0.408	1.273	2.573	**6.823**	3.309
Recall@1: m.i.	20.900	25.071	23.270	26.019	25.403	17.488	8.436	**7.536**	11.896
Recall@1: o.i.	20.616	26.398	22.607	25.972	27.488	22.938	17.204	**8.957**	15.071
Recall@3: m.i.	59.526	73.128	65.592	75.545	76.351	49.573	30.000	**18.957**	34.597
Recall@3: o.i.	58.199	73.886	63.886	72.559	75.498	49.431	38.009	**17.251**	31.706
Recall@5: m.i.	78.673	97.773	87.583	98.626	98.578	66.303	45.498	**27.299**	45.924
Recall@5: o.i.	73.791	97.441	83.081	98.531	98.389	62.512	48.720	**21.706**	40.758
Recall@10: m.i.	85.972	98.483	91.801	98.863	99.005	74.976	58.483	**35.545**	56.493
Recall@10: o.i.	82.891	98.389	89.668	98.863	98.815	72.370	59.479	**30.332**	51.754
Recall@50: m.i.	94.028	99.005	97.014	99.052	99.194	87.915	78.152	**55.735**	74.123
Recall@50: o.i.	93.128	99.147	95.924	99.194	99.147	87.583	77.393	**51.469**	72.464
Recall@100: m.i.	95.877	99.100	97.867	99.100	99.194	91.659	83.744	**63.934**	80.095
Recall@100: o.i.	95.308	99.194	97.299	99.194	99.147	91.422	82.938	**60.900**	78.531

Table C.3: Evaluation on LFW dataset in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with FaceNet and tested with ResNet_152. Lower recall and a higher percentage mean better privacy. The generative methods with two embeddings (see Table C.5) do obtain better results, showing that using multiple embeddings increases robustness and transfer. We can also see that adding Fawkes on top of generative methods improves image privacy, as in the methods StyleGAN + Fawkes (S+F) and VQGAN + Fawkes (V+F).

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512	VQGAN _0.04_128
Percentage	**7.849**	4.494	4.107	2.626	3.470	3.807
Recall@1: m.i.	**2.844**	6.066	6.730	8.152	6.919	6.019
Recall@1: o.i.	**4.597**	8.294	10.047	11.706	9.479	9.431
Recall@3: m.i.	**9.242**	15.640	18.009	22.701	19.242	17.536
Recall@3: o.i.	**9.668**	17.156	19.479	25.545	20.995	19.431
Recall@5: m.i.	**13.175**	22.322	25.166	32.275	27.062	24.787
Recall@5: o.i.	**13.081**	22.701	25.118	32.749	27.109	25.592
Recall@10: m.i.	**18.578**	31.564	35.118	43.175	37.725	34.408
Recall@10: o.i.	**17.725**	30.758	34.360	43.412	36.019	33.460
Recall@50: m.i.	**37.441**	54.028	57.062	67.725	60.142	57.441
Recall@50: o.i.	**35.213**	52.464	55.118	66.019	58.910	55.592
Recall@100: m.i.	**48.531**	64.882	68.436	77.109	71.137	68.104
Recall@100: o.i.	**46.351**	63.934	65.735	75.640	69.147	67.014

Table C.4: Evaluation on LFW dataset in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with FaceNet. Lower recall and a higher percentage mean better privacy. In this case, generative methods optimised with two embeddings obtain worse results than generative methods optimised with only one embedding method in Table 4.1: this is, however, not a fair comparison because we do not use FaceNet for optimisation in this experiment, while we do in Table 4.1.

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512	VQGAN _0.04_128
Percentage:	**5.656**	2.671	2.296	1.549	2.097	2.605
Recall@1: m.i.	**4.976**	8.294	11.422	12.322	9.526	7.630
Recall@1: o.i.	**9.289**	15.450	18.152	18.957	16.635	15.403
Recall@3: m.i.	**15.829**	28.578	32.986	37.488	31.043	28.057
Recall@3: o.i.	**19.763**	34.597	39.100	44.455	38.863	35.545
Recall@5: m.i.	**23.507**	41.327	46.445	55.687	46.161	41.469
Recall@5: o.i.	**26.114**	43.175	49.858	56.303	48.246	43.981
Recall@10: m.i.	**34.218**	52.796	58.957	68.673	59.716	54.929
Recall@10: o.i.	**34.313**	52.796	60.616	68.531	60.284	55.071
Recall@50: m.i.	**56.398**	74.123	78.104	84.692	79.763	76.540
Recall@50: o.i.	**54.929**	73.507	79.005	84.218	79.242	77.109
Recall@100: m.i.	**64.882**	81.611	84.360	89.242	85.118	82.986
Recall@100: o.i.	**63.839**	80.853	84.550	88.863	85.118	83.081

Table C.5: Evaluation on LFW dataset in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet and recognition is tested with ResNet_152. Lower recall and a higher percentage mean better privacy. In this case, generative methods optimised with two embeddings obtain better results than generative methods optimised with only one embedding method, as in Table C.2.

	StyleGAN	StyleGAN _0.02_500	StyleGAN _0.02_1000	VQGAN	VQGAN _0.03_512	VQGAN _0.04_128
Percentage	**11.993**	8.305	7.534	6.067	7.164	8.181
Recall@1: m.i.	**2.038**	4.123	4.550	6.967	4.882	4.739
Recall@1: o.i.	**3.555**	5.687	7.488	7.867	6.540	6.351
Recall@3: m.i.	**6.588**	11.754	12.370	15.640	13.270	11.232
Recall@3: o.i.	**6.919**	12.417	14.787	16.398	13.791	12.038
Recall@5: m.i.	**9.005**	16.209	17.109	20.758	19.052	16.588
Recall@5: o.i.	**9.526**	16.493	18.673	21.611	18.578	16.066
Recall@10: m.i.	**13.081**	22.844	24.976	29.384	26.066	22.417
Recall@10: o.i.	**14.171**	22.938	25.450	29.810	25.450	22.322
Recall@50: m.i.	**27.299**	41.374	43.270	51.280	44.455	41.611
Recall@50: o.i.	**28.863**	41.991	45.024	51.754	46.730	42.464
Recall@100: m.i.	**36.445**	50.806	52.986	61.137	54.360	51.232
Recall@100: o.i.	**38.910**	53.128	56.161	62.796	57.867	52.796

Table C.6: Evaluation on LFW dataset, in the case of transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with ArcFace. Lower recall and a higher percentage mean better privacy. In this case, generative methods optimised with two embeddings obtain better results than generative methods optimised with only one embedding method, as in Table C.1.

	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	0.697	3.975	8.312	0.715	**12.268**	1.633
Recall@1: m.i.	23.571	10.712	5.055	23.831	**3.450**	15.868
Recall@1: o.i.	24.012	15.125	8.064	24.534	**6.239**	16.429
Recall@3: m.i.	70.271	26.359	12.778	66.680	**8.947**	45.256
Recall@3: o.i.	71.274	32.177	15.527	66.800	**12.237**	46.319
Recall@5: m.i.	91.775	36.169	17.934	87.964	**13.340**	60.522
Recall@5: o.i.	91.555	39.860	20.040	84.875	**15.787**	60.923
Recall@10: m.i.	94.965	45.537	24.835	91.575	**18.495**	68.867
Recall@10: o.i.	94.905	48.265	26.239	89.950	**20.963**	69.829
Recall@50: m.i.	97.733	65.597	42.508	96.309	**34.483**	83.470
Recall@50: o.i.	97.553	66.941	45.436	95.125	**36.349**	84.554
Recall@100: m.i.	98.235	72.457	51.775	97.151	**42.989**	88.465
Recall@100: o.i.	98.195	75.125	55.486	96.670	**46.098**	89.749

Table C.7: Evaluation on CC dataset in the case of a transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with FaceNet. Lower recall and a higher percentage mean better privacy. We can see that generative methods' evaluation results for the CC dataset are very similar to the ones for the LFW dataset.

	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	2.880	14.337	**27.089**	2.507	22.651	4.201
Recall@1: m.i.	23.390	2.086	0.562	24.674	**0.301**	19.097
Recall@1: o.i.	21.926	5.918	**1.043**	24.995	1.886	17.031
Recall@3: m.i.	62.708	5.998	**1.143**	67.462	1.484	52.277
Recall@3: o.i.	63.129	10.973	**2.187**	67.683	3.992	49.107
Recall@5: m.i.	78.335	9.468	**1.805**	84.393	2.768	64.654
Recall@5: o.i.	78.154	13.561	**2.949**	83.952	5.296	63.149
Recall@10: m.i.	81.846	14.845	**3.230**	86.399	4.975	70.351
Recall@10: o.i.	82.046	17.553	**4.794**	86.219	7.763	68.706
Recall@50: m.i.	87.182	30.211	**9.007**	89.709	14.624	80.100
Recall@50: o.i.	87.603	31.033	**11.013**	89.749	16.309	78.495
Recall@100: m.i.	89.087	38.736	**14.203**	90.853	20.702	84.092
Recall@100: o.i.	89.348	39.238	**15.226**	90.913	22.287	82.247

Table C.8: Evaluation on CC dataset, in the case without transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with MagFace. We can see that generative methods' evaluation results for the CC dataset are very similar to the ones for the LFW dataset.

	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	2.339	7.232	12.125	1.739	**14.657**	4.074
Recall@1: m.i.	21.184	9.188	5.115	24.453	**4.173**	15.266
Recall@1: o.i.	21.966	12.217	**6.640**	24.955	7.442	10.532
Recall@3: m.i.	55.426	22.628	11.635	63.71	**9.910**	37.733
Recall@3: o.i.	57.392	25.436	**13.260**	64.975	14.002	28.706
Recall@5: m.i.	69.468	28.445	15.165	78.034	**12.839**	48.004
Recall@5: o.i.	70.973	32.016	**16.830**	78.656	17.212	37.813
Recall@10: m.i.	75.165	35.426	19.960	81.825	**17.854**	55.366
Recall@10: o.i.	76.209	39.699	**22.487**	82.648	23.049	44.433
Recall@50: m.i.	84.072	52.879	35.807	88.104	**30.993**	70.090
Recall@50: o.i.	85.035	58.295	40.040	88.546	**38.716**	61.765
Recall@100: m.i.	87.442	61.204	44.714	90.913	**38.837**	76.670
Recall@100: o.i.	88.185	66.419	48.967	90.973	**46.620**	69.087

Table C.9: Evaluation on CC dataset in the case of a transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with ArcFace. We can see that generative methods' evaluation results for the CC dataset are very similar to the ones for the LFW dataset.

	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	6.513	12.668	**20.632**	5.200	15.798	7.708
Recall@1: m.i.	20.542	7.763	**1.224**	24.313	3.972	14.664
Recall@1: o.i.	19.719	12.638	**5.155**	24.875	8.004	12.919
Recall@3: m.i.	53.440	17.673	**3.852**	60.100	9.850	36.871
Recall@3: o.i.	53.902	24.433	**10.251**	59.880	16.108	27.663
Recall@5: m.i.	63.952	22.648	**5.436**	70.812	13.159	44.835
Recall@5: o.i.	64.614	30.451	**12.417**	70.933	20.662	32.397
Recall@10: m.i.	67.643	29.027	**8.666**	74.022	18.415	50.973
Recall@10: o.i.	68.826	37.051	**16.510**	74.223	27.061	36.429
Recall@50: m.i.	74.564	43.771	**19.278**	79.539	33.280	64.092
Recall@50: o.i.	75.206	53.039	**29.027**	79.819	43.290	45.456
Recall@100: m.i.	77.733	51.013	**26.820**	82.006	41.204	69.649
Recall@100: o.i.	78.134	59.980	**37.131**	82.247	51.715	50.291

Table C.10: Evaluation on CC dataset without transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace and MobileFaceNet, and recognition is tested with MobileFaceNet. Lower recall and a higher percentage mean better privacy. We can see that generative methods evaluation results for the CC dataset are very similar to the ones for the LFW dataset.

	VQGAN _0.003_128	VQGAN	VQGAN _0.04_4096	Fawkes	StyleGAN _0.02_1000	AMT-GAN
Percentage	3.025	4.970	6.915	2.652	**7.650**	4.518
Recall@1: m.i.	20.582	12.758	**7.783**	24.012	8.345	16.790
Recall@1: o.i.	20.602	14.584	11.033	24.754	**10.732**	15.908
Recall@3: m.i.	56.951	32.217	**20.361**	61.685	21.846	39.398
Recall@3: o.i.	56.971	33.079	**22.949**	62.347	24.092	34.303
Recall@5: m.i.	69.930	40.943	**27.021**	75.165	28.646	49.328
Recall@5: o.i.	69.629	41.083	**28.686**	74.905	30.150	41.805
Recall@10: m.i.	74.463	49.328	**35.065**	78.816	36.489	56.289
Recall@10: o.i.	74.183	49.589	**36.189**	78.696	37.232	47.141
Recall@50: m.i.	82.608	66.379	**53.521**	85.496	54.644	71.414
Recall@50: o.i.	82.508	67.041	**54.664**	85.216	56.851	59.920
Recall@100: m.i.	85.657	73.561	62.628	87.823	**62.327**	77.553
Recall@100: o.i.	85.537	73.400	**63.591**	87.803	64.754	65.236

Table C.11: Evaluation on CC dataset in the case of a transfer to another embedding: VQGAN and StyleGAN are optimised with MagFace + MobileFaceNet, and recognition is tested with ResNet_152. Lower recall and a higher percentage mean better privacy. We can see that generative methods' evaluation results for the CC dataset are very similar to the ones for the LFW dataset.