Quantifying Likeness: A Simple Machine Learning Approach to Identifying Copyright Infringement in (AI-Generated) Artwork

Michaela Drouillard [*1] **Ryan Spencer** [*1] **Nikée Allen** [1 2] **Tegan Maharaj** [1 3]

Abstract

This study proposes an approach aligned with the legal process to quantify copyright infringement, via stylistic similarity, in AI-generated artwork. Copyright infringement by AI systems is a topic of rapidly-increasing importance as generative AI becomes more widespread and commercial. In contrast to typical work in this field, and more in line with a realistic legal setting, our approach quantifies similarity of a set of potentially-infringing "defendant" artworks to a set of copyrighted "plaintiff" artworks. From a machine learning perspective, this is a straightforward image classification task which can be accomplished in a quite simple, low-resource setting. We present a case-study using our approach with Mickey Mouse as the plaintiff, and perform a thorough hyperparameter search and robustness analysis. The aims of this work are to illustrate the potential of the approach, and identify settings which generalize well, such that it is as 'plug and play' as possible for artists or legal experts to use with their own plaintiff sets of artworks.

1. Introduction

The rapid development and widespread use of generative AI models has raised concerns among creators about potential job disruption and copyright infringement. In the courts, these technologies are challenging our understanding of how legal concepts like "substantial similarity" and "fair use" apply within the generative AI supply chain (Lee et al., 2023).

[*]Equal contribution [1]Faculty of Information, University of Toronto, Toronto, Ontario, Canada [2]Faculty of Law, University of Toronto, Toronto, Ontario, Canada [3]Schwartz Reisman Institute for Technology and Society. Correspondence to: Michaela Drouillard <michaela.drouillard@mail.utoronto.ca>, Ryan Spencer <r.spencer@mail.utoronto.ca>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

While there has been significant machine learning research on copyright detection, much of it focuses on identifying *if any* copyrighted works are detected, rather than offering tools for understanding *how* copyright might be being violated. More specifically, the problem is typically framed as recognizing whether a set of artwork(s) are present in or substantially similar to any data in the training corpus. This is a difficult problem, ideally requiring white-box access to the training data and entire training procedure. We note that this is not representative of the way copyright infringement cases are typically dealt with in court. To address this, we propose a novel approach using a small, customizable model that artists can use to assess the likelihood of AI-generated work infringing on their work. This approach reframes the machine learning problem to be more in line with (and thereby usable in) legal proceedings: We identify a **plaintiff set** of artworks which are copyrighted, and a **defendant set**, for example AI-generated ones, which are potentially infringing. Our approach consists of training a machine learning model to correctly classify the plaintiff artworks among a set of similar artworks, and performing inference with this model on the defendant set, using the softmax-normalized probabilities of the plaintiff class as similarity scores.

Framing the quantification of substantial similarity as a classification problem among expert-identified sets has many advantages: (1) it makes use of expert knowledge and human judgement in assessing which classes are relevantly similar; (2) it thus allows the effective capacity of the model to be focused on features of the plaintiff class that are relevant for distinguishing it from similar artwork, rather than spread across the many possible categories if we were to consider the entire training corpus; (3) we can therefore use relatively low-resource models, accessible to non-experts.

Using computer vision, our model identifies the extent to which an image replicates the likeness of copyrighted entities. The model outputs softmax-normalized probabilities, and these probabilities serve as similarity scores, with higher probabilities indicating greater similarity to the copyrighted work. Key features contributing to these scores are highlighted in visualizations. We use saliency mapping,

feature mapping, and high-similarity template matching visualizations, which allow users to interpret the similarity scores in conjunction with legal and artistic experts.

The small size and customizability of our model are key features that make it practical and accessible for potential plaintiffs, including small studios and artists. The intended use case is for artists with a full corpus of copyrighted material to have a quantitative assessment of the substantial similarity that a potentially infringing AI-generated work bears to their own style. It is important to note the similarity score is not intended and should not be used as replacement for expert legal judgment. On the contrary, its purpose is to provide a narrow, specific support to arguments that rest on substantial similarity, which fits into the broader context of a case with qualitative and other arguments.

Rather than trying to automate expertise, we propose building a support tool with experts in the loop. The strength of quantitative methods is that they force us to make our understandings and usages of concepts explicit enough than they can be falsified, and, if needed, redefined. This study suggests one method for making concepts like likeness or substantial similarity explicit – we offer it as contribution to broader framework that combines machine learning, legal expertise, scholarship and artists' interests in order to establish clearer guidelines for intellectual property law in a time when authorship, creation, and inspiration are being renegotiated.

2. Related Work

Ever since the rise of social media platforms and the mass growth of avenues for image sharing, there has also been a steady growth in the development of technological solutions to detect and manage copyright breaches. Google's Content ID automatically identifies and manages copyrighted audio and visual files on Youtube (Google, 2023). Kim et al. proposes a photo copyright identification framework that accurately identifies copyright infringements of photos that have been manipulated through techniques like cropping, collages, or color changes(Kim et al., 2021). Beyond building tools themselves, interdisciplinary work utilizing computational concepts to parse copyright issues has also contributed to this trend. For instance, Scheffler et al. (2022) introduces a quantitative framework for assessing "substantial similarity" in copyright law. This framework leverages computational concepts like description length, inspired by Kolmogorov-Levin complexity, to quantitatively evaluate how derivative works may or may not be similar to their originals. By leveraging an interdisciplinary research lens, it offers a more structured approach to a traditionally subjective area of law by quantifying the "novelty" required to produce a derivative work both with and without access to the original copyrighted elements.

Much of the existing work on copyright detection tackles a difficult problem of what we term **general similarity detection**, which includes problems like detecting exact copies of copyrighted work, or detecting copies using the logits of pre-trained models. This kind of work hasn't been accessible to anyone other than machine learning experts because it's a complex problem. We frame a different and more tractable problem of **local similarity detection**, which mirrors the real world setting where we know the reference class we're differentiating from, and enables us to use very small models that can easily and cheaply be trained by non-experts.

An example of this differentiation is comparing our framework with CLIP, which is trained on diverse internet-collected image-text pairs, and assesses text-to-image similarity by comparing embeddings from both modalities (Radford et al., 2021). Our approach differs from this broader approach that we fine-tune a model on context-relevant images, using the softmax score as a similarity metric, which can then be used as a support tool for those defending their work.

In qualitative research, there's been a growing discussion in how to attribute agency and authorship, and thereby infringement, in intellectual property law. One challenge is maintaining a consistent definition of these concepts across different mediums. Alexandre Montagu and David Bellos, in their book *Who Owns this Sentence?*, provide a cultural, legal, and global history of the idea of copyright, explaining the concept of fair use and the difficulties in defining and applying it consistently across various contexts like music, art, and AI-generated content (Bellos & Montagu, 2024).

Considering the nuances of agency in the process, Ginsburg and Budiardjo propose a model of authorship based on conception and execution, concluding that even the most advanced machines are merely agents of the humans who design or use them (Ginsburg & Budiardjo, 2019). Beyond authorship, there are also difficulties in defining and applying copyright consistently throughout the AI supply chain specifically, as demonstrated by Lee et al. (2023). In this context, our work targets the "generation" phase of this supply chain, as opposed to building a tool that identifies infringing objects within the training data or analysing the prompt as the infringing object.

Finally, the recent lawsuits filed by artists Sarah Andersen, Kelly McKernan, and Karla Ortiz and several other artists against Stability AI, Midjourney, and other companies using Stability AI's Stable Diffusion models will likely set important precedents for how intellectual property law is applied to AI systems. The plaintiffs contend that the defendants have unlawfully used their copyrighted works to train models without authorization. The United States District Court, in its tentative rulings from May 2024, decided

not to dismiss the direct and induced copyright infringement claims, suggesting that courts may be open to offering greater protections to copyright holders whose works are used to train AI models without prior consent (United States District Court for the Northern District of California, 2024). Our research engages with the assumptions, theories and challenges underpinning these ongoing legal battles and contributes to the development of clearer guidelines and support tools for determining copyright infringement in AI-generated content.

3. Legal Context

The concept of "substantial similarity" is central to determining copyright infringement, but there is no bright-line rule for establishing it. Courts often consider factors such as the "total concept and feel" of the works in question and the level of creativity involved in the copyrighted work (first defined in (U.S. Court of Appeals for the Ninth Circuit, 1977)), in conjunction with expert testimony and analysis.

To establish copyright infringement, a plaintiff must first demonstrate that the alleged infringer actually used the copyrighted work in their purportedly infringing activities. Sometimes plaintiffs have direct evidence that the alleged infringer used their copyrighted work in the defendant's purportedly infringing activities. For instance, a defendant may admit that the copyrighted work was their inspiration in creating their own work. Or perhaps the plaintiff can point to eyewitnesses of the alleged copying. But often, perhaps typically, direct evidence is lacking. When it is lacking, courts may consider a combination of (1) evidence of the defendant's access to the copyrighted work; and (2) similarities between the defendant's work and the original copyrighted work that suggests copying, in determining whether the alleged infringer actually copied from the copyrighted work.

Two works are substantially similar when "the ordinary observer, unless [they] set out to detect the disparities, would be disposed to overlook them, and regard their aesthetic appeal as the same" (U.S. Court of Appeals for the Second Circuit, 1960). A common test is a "holistic, subjective comparison of the works to determine whether they are substantially similar in total concept and feel" (U.S. Court of Appeals for the Ninth Circuit, 2018).

Historically, some courts have dispensed with the requirement for evidence of access if the works are so "strikingly similar" that it is more likely than not that copying occurred (U.S. Court of Appeals for the Second Circuit, 1946). Interestingly, the Ninth Circuit has recently retired the related inverse ratio rule - the concept that as evidence of access increases, the evidentiary threshold for identified similarity to prove copying decreases, and vice versa. In light of this change, it is possible that other circuits may follow suit in the future to maintain consistency in jurisprudence relating to copyright.

When assessing substantial similarity, courts often employ the "extrinsic-intrinsic test," which was first articulated in *Sid Marty Krofft Television Productions, Inc. v. McDonald's Corp*. (U.S. Court of Appeals for the Ninth Circuit, 1977). The extrinsic component of the test involves an objective analysis of the similarities in ideas and expression between the works, while the intrinsic component is a more subjective assessment of overall similarities from the perspective of the "ordinary reasonable person" (or a similar description of a reasonable individual possessing no related expert knowledge) (See U.S. Court of Appeals for the Ninth Circuit (2004); U.S. Court of Appeals for the Ninth Circuit (1994); U.S. Court of Appeals for the Ninth Circuit (1986)). Although expert testimony may be considered in the extrinsic analysis, it is inappropriate for the intrinsic test due to its focus on the perspective of the ordinary person (U.S. Court of Appeals for the Ninth Circuit (1988); U.S. Court of Appeals for the Ninth Circuit (2016)). Further, the application of the substantial similarity test may vary depending on the subject matter and medium of the works in question.

By providing a quantitative assessment of substantial similarity, frameworks like ours can aid plaintiffs in defending their intellectual property rights by offering explicit similarity metrics, tailored to individual contexts. In the future, combining computational tools with interpretability methods may make it possible to identify where infringement occurs in the training and generation process, thereby identifying a new kind of "infringing object", which would advance our understanding of how to apply copyright protection in the context of generative AI.

That being said, these tools should not be viewed as a replacement for expert analysis and legal judgment. The concept of substantial similarity is inherently complex and context-dependent, and courts have emphasized the importance of considering the "total concept and overall feel" of the works in question, rather than relying on mechanical dissection or quantitative measures alone (U.S. Court of Appeals for the Ninth Circuit, 1977; 2018). Further, these tools are not sufficient to support the practice of law without a relevant education or bar membership.

4. Experiments

In our experiments, we first present a basic model (referred to throughout this paper as "basic model") that assesses the similarity of images generated by Claude and DALL-E to stills of the 1928 Steamboat Willie version of Mickey

Mouse. We train this model using three classes: Winnie the Pooh, Steamboat Willie, and Donald Duck. We include feature visualizations that identify where the model was able to differentiate between style and content (this dichotomy, which is foundational in art criticism, also maps loosely on the legal concepts of ”idea” and ”expression”. Content refers to what the image depicts, style refers to the way in which it is depicted).

Iterating on our basic model, we introduce several improvements: first, we experiment with data augmentation techniques using the basic model's training dataset. Then, we use a different training dataset which includes both more classes, and includes classes of characters that are more visually similar to Mickey Mouse (in other words, more mouse cartoons, rather than ducks and bears). Once we collected this dataset and performed initial runs, we experimented with different combinations of batch sizes and weight decay values, different learning rates, and cross-validation. Then, we perform an analysis of the synthetic data again and output the similarity scores using the optimal model hyperparameters.

4.1. Basic Model

We fine-tuned the fully connected (FC) layer of a ResNet-18 model, chosen for its superior performance and computational efficiency compared to AlexNet and VGG. The ResNet-18 model consists of 18 layers, including convolutional, max-pooling, and a fully connected layer with 512 features. It takes a 224x224 input image and outputs predictions for each class.

The ResNet architecture's deep residual learning framework helps alleviate the gradient degradation problem encountered in earlier models like AlexNet and VGG (He et al., 2015). Fine-tuning only the FC layer while keeping the pre-trained convolutional layers frozen was based on previous literature which suggests that pre-trained classification layers are necessary for better optimization during the fine-tuning process and increasing network depth (Shermin et al., 2019). This was confirmed by trial runs, which yielded higher accuracy and fewer false positives compared to fine-tuning all layers.

4.1.1. Basic Model Data

The training set consists of 1026 images across three classes: 565 images of Mickey Mouse, 278 of Donald Duck, and 183 of Winnie the Pooh with random resizing, random cropping, random horizontal flip, random affine, random rotation, colour jitter, random perspective, and random erasing transformations applied during preprocessing. We will refer to these transformations as ”naive transformations” throughout the paper (in contrast with the AugMix transformations in 4.2; see A.1 for exact naive transformation values). The Mickey Mouse images were drawn from stills of Steamboat Willie (1928) and Gallopin' Gaucho (1928), two works that entered the public domain this year. For the model classification tasks, we created a dataset of 50 images of cartoon mice with varying degrees of likeness to Steamboat Willie using DALL-E and generated prompts from the Claude API.

4.1.2. Model Set-Up

The model was trained using the cross-entropy loss function and RMSprop optimizer, a base learning rate of 0.0001, a batch size of 10 and a weight decay of 0.1, with L2 regularization. The model was trained for 100 epochs with a 70% training and 30% validation split using a GPU environment with CUDA, PyTorch, and TensorFlow environments.

4.1.3. Results

The results of the model performance are summarized in Table 1. The model demonstrated strong performance metrics, achieving a training loss of 0.1152 with an accuracy of 96.09%, and a validation loss of 0.0558 with an accuracy of 98.70%. The model reached its best validation accuracy at 99.21%.

Metric	Train	Validation
Loss	0.1152	0.0558
Accuracy	0.9609	0.9870
Best Validation Accuracy	0.9921	

Table 1: Performance metrics of the basic model on the training and validation datasets.

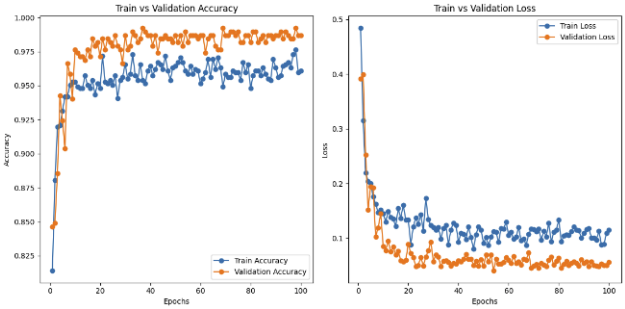


Figure 1: Model training accuracy plotted against validation accuracy

4.1.4. Feature Visualizations

We used three visualization techniques to interpret our model's decision-making: feature mapping, saliency maps, and template matching (using Figure 2 as a reference image).

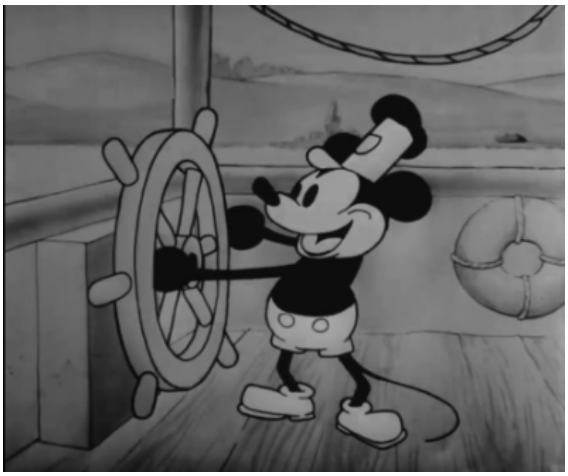
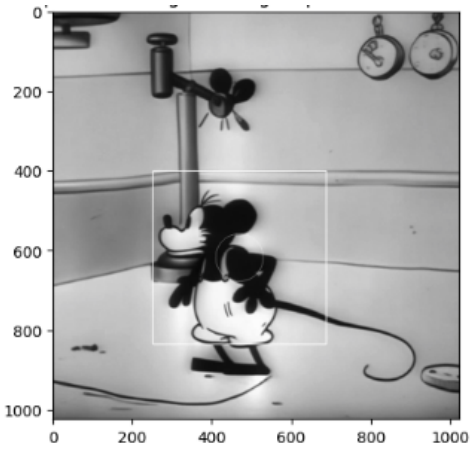


Figure 2: Steamboat Willie (1928)

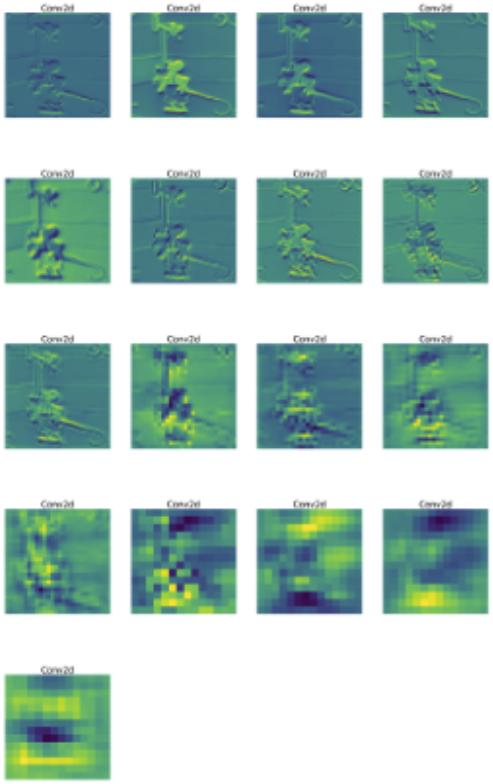
From our outputs, we selected Output 1 (see Figures 3a, 3b, and 3c) and Output 2 (see Figures 4a, 4b, and 4c) for this report since they differ significantly in similarity to Steamboat Willie. The model identified Output 1 as having a 0.999 probability of belonging to the Mickey Mouse class, suggesting that this image is very likely to be reproducing the likeness of a copyrighted image. In Figure 3a and in Figure 3b, the importance is placed on the style of the ears, tail, and the shape of the cartoon's body. The pants are also a nearly exact match to Steamboat Willie's style of pants, which is highlighted by the saliency map.

The model identified Output 2 as having a 0.021 probability of belonging to the Mickey Mouse class. In Figure 3a, the only important feature that aligns with a Steamboat Willie reference image is the shape of the cartoon's ear. This inadvertently suggests a strong model performance — the mouse is surrounded by polygons very similar to the ear, and yet the model was able to distinguish between a polygon and a polygon that is part of an ear.

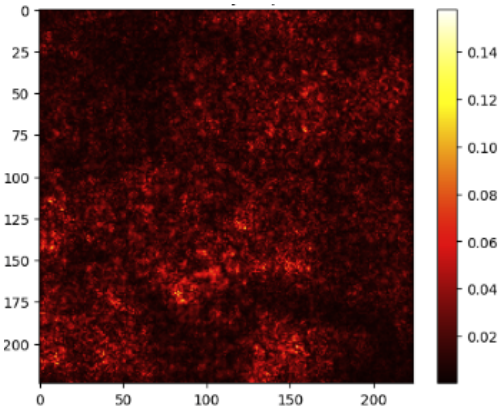
These results demonstrate that the model was successful in separating style and content when identifying the likeness of a cartoon figure in generative AI outputs. The model contextualized the polygons in Output 2 as belonging to an ear but did not classify the image as strongly similar to Steamboat Willie based on this content recognition alone. In Output 1, the style, remarkably similar to 1920s-era Mickey Mouse aesthetics, led to a very high probability score of reproducing likeness.



(a) Template Matching applied to Output 1

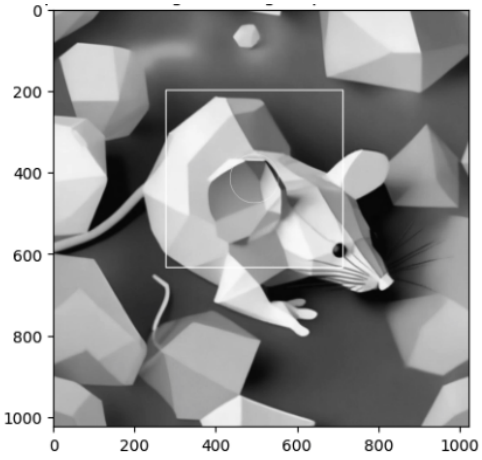


(b) Feature Map for Output 1

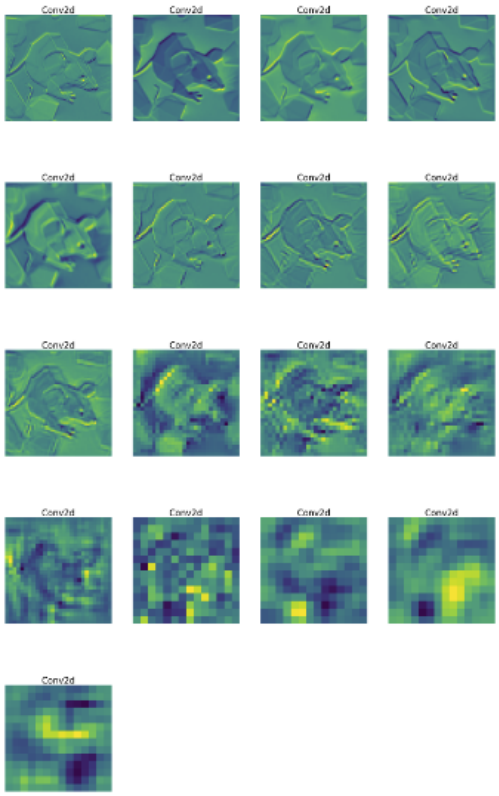


(c) Saliency Map for Output 1

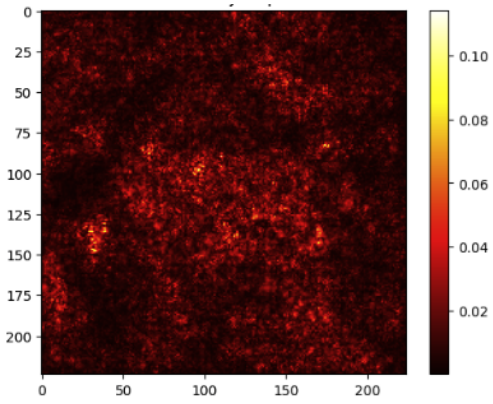
Figure 3: Visualizations for Basic Model Output 1



(a) Template Matching applied to Output 2



(b) Feature Map for Output 2



(c) Saliency Map for Output 2

Figure 4: Visualizations for Basic Model Output 2

4.2. Robustness Experiment

Before using different classes in the training data, we experimented with implementing AugMix transformations to our basic model. AugMix, which was developed in Hendrycks et al. (2020), combines simple augmentation operations with a consistency loss based on Jensen-Shannon Divergence. By mixing multiple augmented images, AugMix generates diverse yet semantically consistent transformations, which helps models withstand unforeseen data corruptions while maintaining high performance on clean data (see our code implementation in A.2). We applied AugMix both with and without the original naive transformations as well, and ultimately found that the model was more prone to overfitting when AugMix was applied. This suggests that it may need to be modified, or may not be as effective for smaller models and datasets.

4.2.1. Model Set-Up

We used the basic model (4.1.2), with the same data (4.1.1, for the augmentation experiments. Our implementation of the AugMix framework includes three main components (see A.2 for code): 1) Augmentation operations: We implemented random rotation, horizontal flipping, and color jittering. Each operation's intensity is controlled by a severity parameter.; 2) Augmentation chain: Multiple augmentations are applied sequentially to create diverse transformations. The number of operations in each chain is randomly chosen between 1 and 4.; 3) Mixing strategy: We generate 3 augmented versions of each image and mix them using weights sampled from a Dirichlet distribution. The final mixed image is then combined with the original image using a beta distribution.

We incorporated a Jensen-Shannon Divergence (JSD) consistency loss to encourage consistent predictions across different augmentations of the same image. This loss is added to the standard cross-entropy loss during training, with weights of 2, 12 or 20 in our experiments.

The trial runs included: 1) running the model with only AugMix transformations with JSD loss weights of 2, 12 and 20, and then 2) running the model with both the naive transformations and the AugMix transformations applied, with JSD loss weights of 2 and 12.

4.2.2. Results

Overall, we found that using AugMix transformations did not outperform the naive transformations in the basic model. The best performance we observed was with applying both the naive and AugMix transformations, and using a JSD loss of 2. Figures 5 displays moderate overfitting with a stable performance, although the validation performance is still worse than that observed in Figure 1. See A.3

for complete experiment visualizations.

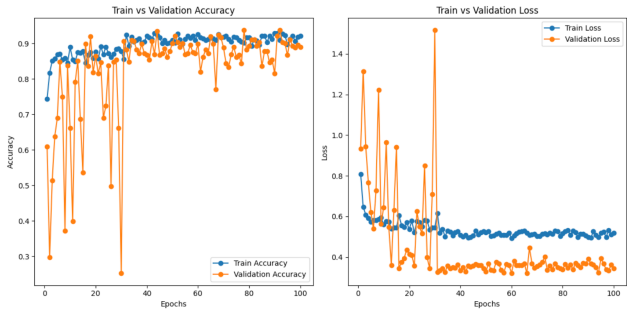


Figure 5: Naive + AugMix + JSD loss 2 performance

4.3. Hyperparameter Tuning

4.3.1. Data

We included more classes, with cartoon characters that are more similar to Mickey than Donald Duck and Winnie the Pooh. Specifically, we introduced a new batch of data comprising characters from Warner Brothers' Foxy (a character designed by former Disney animators Hugh Harman and Rudolph Ising (IMDB, n.d.)), Van Beuren Studios' Milton the Mouse, and Hanna-Barbera's Tom and Jerry (see A.4 for images). Both Foxy and Milton serve as "ground truth" likenesses to Mickey Mouse, as Disney pursued legal action against Van Beuren Studios for Milton the Mouse's infringement on Mickey Mouse's likeness (wal, 1932) (see Figure 6 for an annotated comparison of Milton and Mickey used in the original case (Walt Disney Productions, Ltd. v. Pathe Exchange, Inc. and the Van Beuren Corporation, 1932)). Disney won the case against Milton, while Warner Brothers discontinued Foxy's appearances after three shorts (Beck & Friedwald, 1989). Tom and Jerry are included to evaluate model performance with color images and to understand how other anthropomorphized characters, such as Tom, a cat, influence model performance.

The dataset is relatively small due to the limited number of images available for Foxy and Milton, whose careers were brief. The balanced dataset consists of 235 training images and 117 validation images for each class. Images of Foxy were obtained from the three Merrie Melodies shorts, while images of Milton were sourced from Aesop's Fables. The FFmpeg function was used to extract JPEGs from the public domain animated shorts. We applied the naive data transformations from the basic model to the training data.

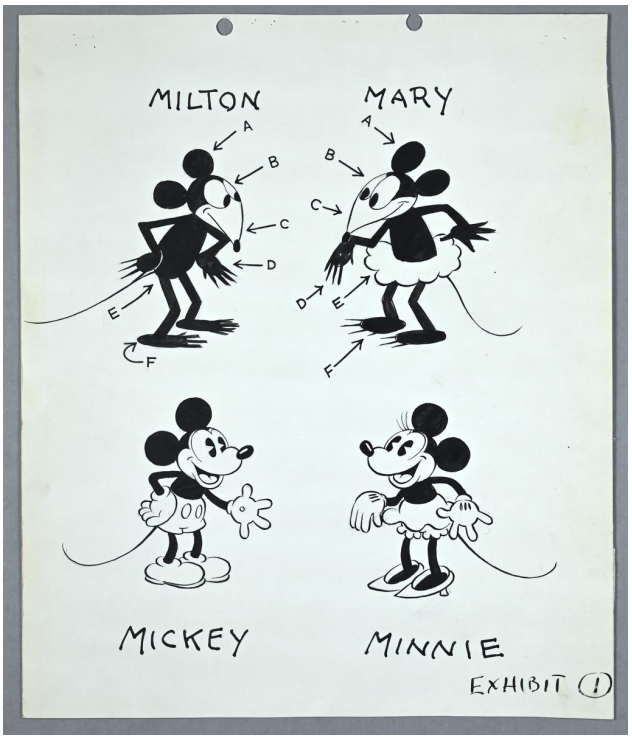


Figure 6: Exhibit 1, annotated comparison of Milton the Mouse and Mickey Mouse, and Rita the Mouse and Minnie Mouse

4.3.2. Results

We applied the following combinations of batch sizes and weight decays to identify the best combination for our model: batch size 4 and weight decay 0.01, batch size 6 and weight decay 0.01; batch size 4 and weight decay 0.001, batch size 6 and weight decay 0.001; batch size 4 and weight decay 0.0001, batch size 6 and weight decay 0.001.

We observed the best performance using a batch size of 4 with a weight decay of 0.001 (see Table 2, and see all combinations classification reports in A.5).

Table 2: Classification report for batch size 4, weight decay 0.001

	precision	recall	f1-score	support
Foxy	0.91	0.92	0.92	117
Jerry	0.81	0.96	0.88	117
Mickey	0.94	0.87	0.90	117
Milton	0.95	1.00	0.97	117
Tom	0.95	0.78	0.85	117
accuracy			0.91	585
macro avg	0.91	0.91	0.91	585
weighted avg	0.91	0.91	0.91	585

4.4. Learning Rate Experiments

Based on the findings by Li et al. (2020) in "Budgeted Training: Rethinking Deep Neural Network Training Under Resource Constraints", we used a Linear Decay Learning Rate Scheduler. The implementation of a Linear Decay Learning Rate was shown to be beneficial for a ResNet model constrained by a fixed resource budget, offering a simple, robust, and high-performing compared to other learning rate schedules (Li et al., 2020). This is crucial for our model, especially since we aim for real-world applications where limited computational resources are available to creative and legal stakeholders. This approach systematically reduces the learning rate in proportion to the total iteration budget, which is especially effective under resource-constrained settings. Compared to our previous runs with StepLR schedulers, the linear decay learning rate has proven to be more effective lowering our validation learning loss and increasing validation accuracy.

A Learning Rate Find was performed to locate an optimal learning rate for the model. From the plot above, a learning rate between 0.001 and 0.01 is ideal. The learning rate has been set at 0.001 for the previous experiments and this plot confirms that value is optimal for our model.

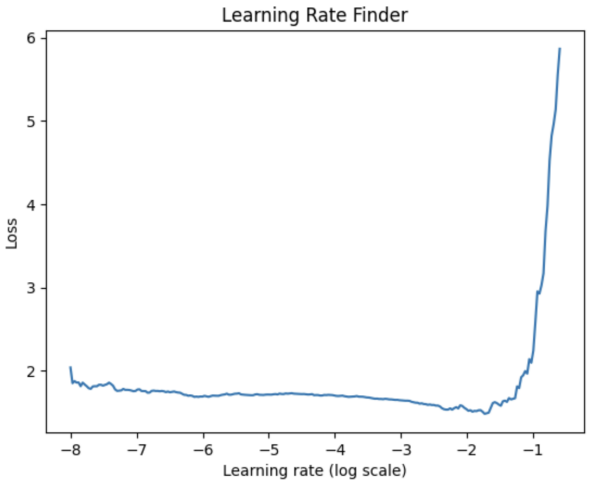


Figure 7: Optimal learning rate range plot.

4.5. Cross-Validation

We performed a 5-fold cross-validation to test our model's validation accuracy and validation learning loss. We found that a batch size of 6 on the fourth fold yielded the best performance, with an accuracy of 0.84, and overall yielded a strong and stable performance that almost matched runs using a batch size of 4. We include both batch sizes 4 and 6 in our analysis, which, given the stochastic nature of each batch size, will allow for more interpretation of accuracies across both batch sizes.

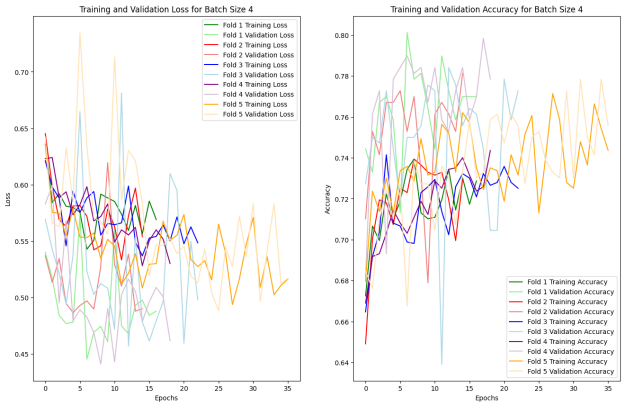


Figure 8: Cross-Validation Performance

4.6. Optimal Run

Based on the results of our experiments, we ran the model with a batch size of 4, a weight decay of 0.001, a learning rate set at 0.001, and applied the naive transformations to the dataset of 5 classes. A linear decay learning rate sched-

uler and an Adam optimizer with AmsGrad are employed with early stopping implemented with a patience of 10. The model ran on a A100 GPU and converged at epoch 27 with a validation learning loss of 0.309603 and a validation accuracy of .91.

Metric	Train	Validation
Loss	0.4669	0.3096
Accuracy	0.8289	0.9060
Best Validation Accuracy	0.9921	

Table 3: Performance metrics of the optimal model on the training and validation datasets.

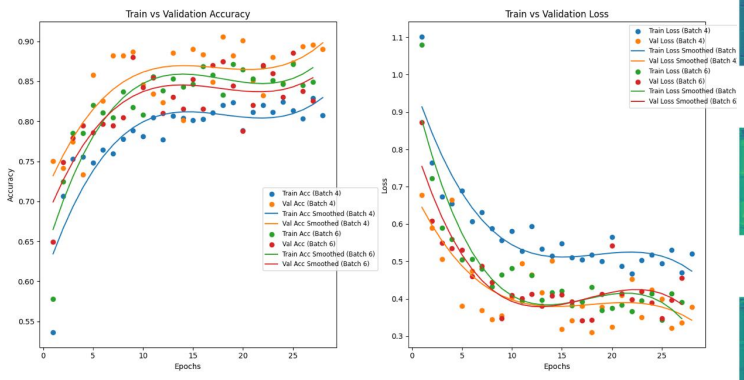
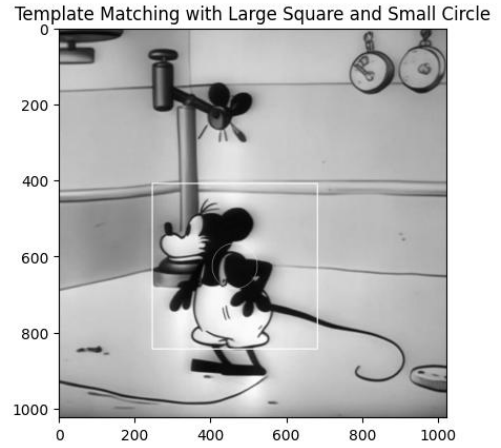


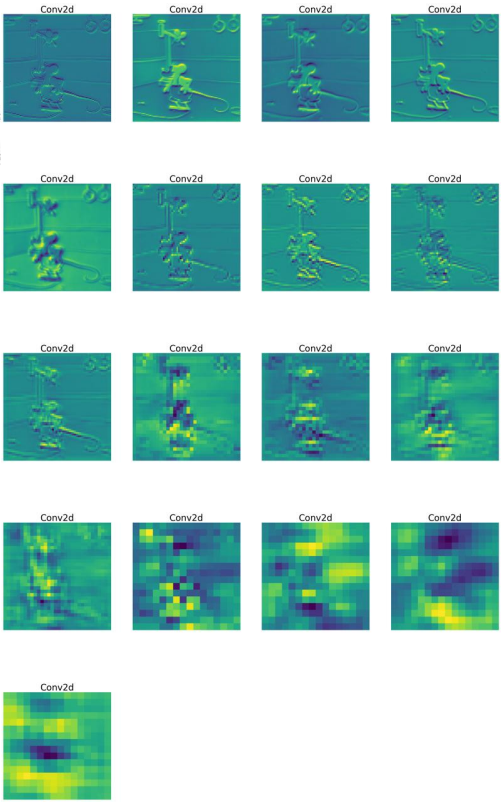
Figure 9: Optimal Run Performance

For Output 1 (see 10), the model output a 0.73 probability that the image belonged to the Mickey class (with a probability of 0.26 that the image belonged in the Foxy class). This shows that the model was able to recognize the "ground truth" of Steamboat Willie-era Mickey Mouse style (which was copied in Foxy as well). Like the feature visualizations in the basic model, the ears, body, tail and nose shape of this image are all identified throughout as being significant indicators of stylistic similarity.

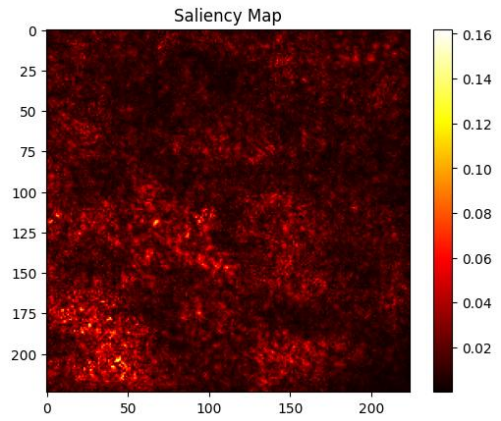
For Output 2 (see 11), the model outputs 0.05 probability that this model belonged to the Mickey class. Figure 11a shows that the model is confounded by the content within the circle, but identifies the torso of the mouse as bearing some resemblance to Mickey (perhaps due to the fact that this is the only area in the image with the stark black and white contrast we see on Mickey's body). The ears, facial expression and feet aren't included in the matching, which is wear the image differs significantly in style to Mickey.



(a) Template Matching

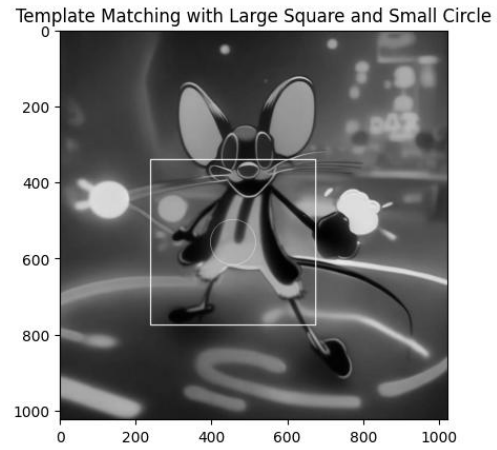


(b) Feature Map

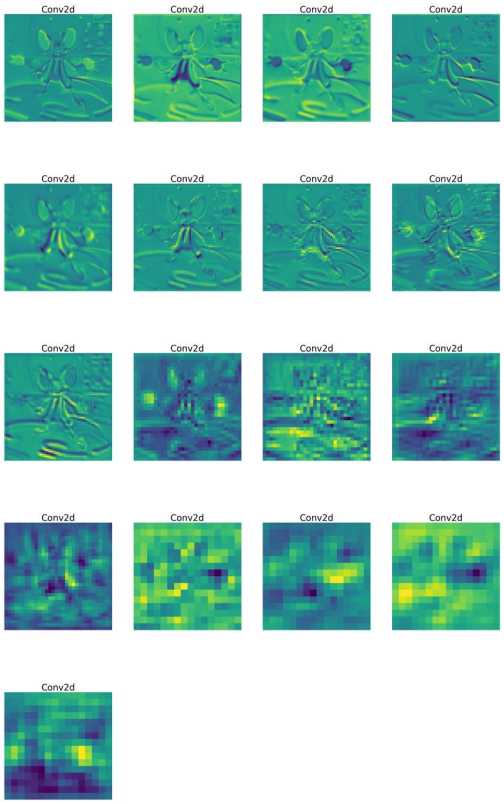


(c) Saliency Map

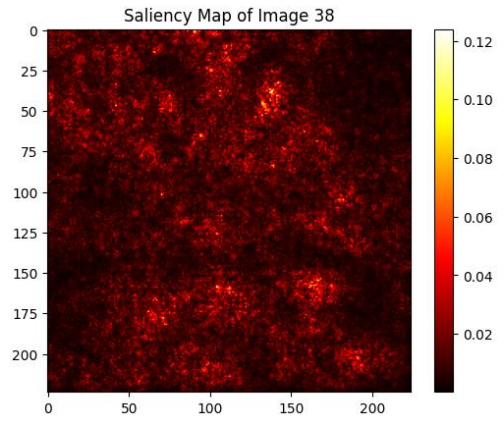
Figure 10: Analyses for Optimal Model Output 1



(a) Template Matching



(b) Feature Map



(c) Saliency Map

Figure 11: Analyses for Optimal Model Output 2

5. Future Work

Since our goal is to publish the model with a plug-and-play user-friendly interface, our next steps include working on interpretable visualizations and ensuring that the model generalizes well to new datasets (for instance, seeing how the model generalizes to Google's Quick, Draw! dataset (Google Creative Lab, 2023)).

We also propose collaborating with artists whose works' likeness have allegedly been reproduced by generative AI models and training the model to classify between original work, generative work, and other works that artists identify as being influential on their own style. By comparing human, computational, art criticism and legal assessments of likeness, we aim to develop a framework for clearer guidelines on determining copyright infringement in AI-generated material, contributing to more robust and consistent standards for evaluating potential infringement.

A. Appendix

A.1. Naive Transformations

```
transformTrain = transforms.Compose([
transforms.RandomResizedCrop(224),
        transforms.RandomHorizontalFlip
            (),
        transforms.RandomRotation(10),
        transforms.ColorJitter(
            brightness=0.2, contrast
            =0.2, saturation=0.2, hue
            =0.1),
        transforms.RandomAffine(0,
            translate=(0.1, 0.1)),
        transforms.ToTensor(),
        transforms.RandomPerspective(
            distortion_scale=0.05, p
            =0.5),
        transforms.RandomErasing(p=0.1,
             scale=(0.02, 0.33), ratio
            =(0.3, 3.3), value=0,
            inplace=False),
        transforms.Normalize([0.485,
            0.456, 0.406], [0.229,
            0.224, 0.225])
    ])
```

A.2. AugMix Code Implementation

```
def random_rotation(image, severity):
  max_angle = 90 * severity / 10
  angle = random.uniform(-max_angle,
      max_angle)
  return image.rotate(angle)
```

```
def random_horizontal_flip(image,
    severity):
  flip_prob = severity / 10
  if random.random() < flip_prob:
    return image.transpose(Image.
        FLIP_LEFT_RIGHT)
  return image

def color_jitter(image, severity):
  factor = severity / 10
  brightness_factor = 1 + (0.2 * factor
      )
  contrast_factor = 1 + (0.2 * factor)
  saturation_factor = 1 + (0.2 * factor
      )
  hue_factor = 0.1 * factor
  return transforms.ColorJitter(
    brightness=brightness_factor,
    contrast=contrast_factor,
    saturation=saturation_factor,
    hue=hue_factor
    )(image)

augmentations = [
    random_rotation,
    random_horizontal_flip,
    color_jitter,
]

def AugMix(image, severity_range=(1,
    10), width=3, depth=-1, alpha=0.5):
    image_pil = to_pil_image(image)

    ws = np.float32(np.random.dirichlet
        ([alpha] * width))
    m = np.float32(np.random.beta(alpha
        , alpha))
    mix = torch.zeros_like(transforms.
        ToTensor()(image_pil))

    for i in range(width):
        image_aug = image_pil.copy()
        d = depth if depth > 0 else np.
            random.randint(1, 4)
        for _ in range(d):
            op = random.choice(
                augmentations)
            severity = np.random.
                uniform(*severity_range
                )
            image_aug = op(image_aug,
                severity)
        mix += ws[i] * transforms.
            ToTensor()(image_aug)

    mixed = (1 - m) * transforms.
        ToTensor()(image_pil) + m * mix
    return mixed

def jsd_loss(p, q, r):
    p_loss = F.kl_div(F.log_softmax(p,
        dim=1), F.softmax((q + r) / 2.,
         dim=1), reduction='batchmean')
    q_loss = F.kl_div(F.log_softmax(q,
        dim=1), F.softmax((p + r) / 2.,
         dim=1), reduction='batchmean')
    r_loss = F.kl_div(F.log_softmax(r,
        dim=1), F.softmax((p + q) / 2.,
         dim=1), reduction='batchmean')
    return (p_loss + q_loss + r_loss) /
         3.
```

A.3. Augmentation Experiment Performances

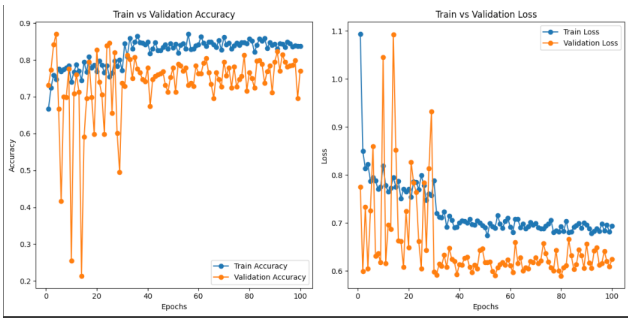


Figure 12: AugMix + JSD loss 12 performance

AugMix + JSD loss 12

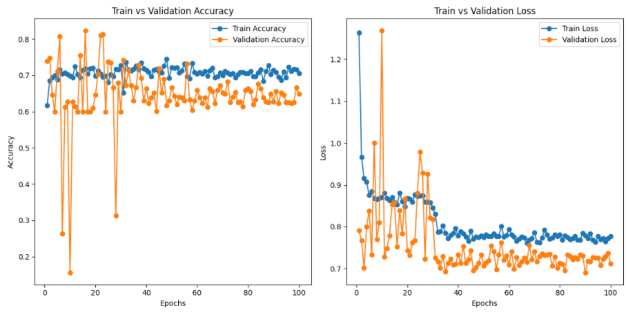


Figure 13: AugMix + JSD loss 20 performance

AugMix + JSD loss 20

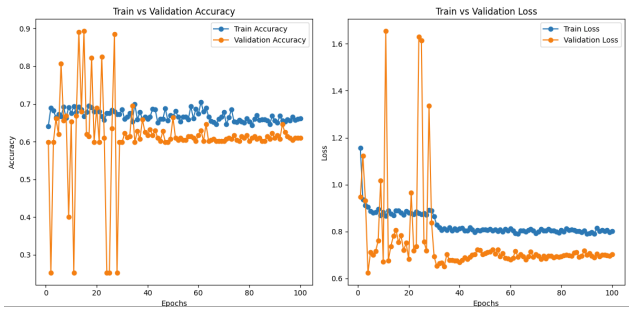


Figure 14: Naive + AugMix + JSD loss 12 performance

Naive + AugMix + JSD loss 12

A.4. Additional Classes

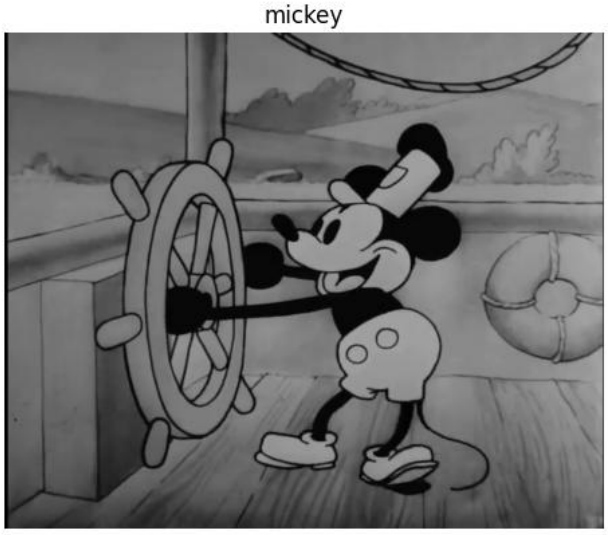


Figure 15: Mickey

Figure 16: milton

Figure 17: Foxy

Figure 18: Tom

Figure 19: Jerry

A.5. Hyperparameter Tuning Experiment

Batch Size 6 with Weight Decay 0.001: We observed similar results to Batch Size 4, however, the precision and recall of the Mickey class are imbalanced compared to Batch Size 4 and we have a lowered overall accuracy of the validation set.

Table 4: Classification report for batch size 6 with Weight Decay 0.001

	precision	recall	f1-score	support
Foxy	0.78	0.95	0.86	117
Jerry	0.80	0.95	0.87	117
Mickey	0.94	0.69	0.80	117
Milton	0.95	1.00	0.97	117
Tom	0.94	0.76	0.84	117
accuracy			0.87	585
macro avg	0.88	0.87	0.87	585
weighted avg	0.88	0.87	0.87	585

Batch Size 4 with Weight Decay 0.0001: We observed that while the accuracy remained constant to Batch Size 4 with a Weight Decay of .001, a concern of overfitting remains due to the small size of the data set, therefor a more robust weight decay of .001 is preferred moving forward.

Table 5: Classification report for batch size 4 and weight decay of 0.0001

	precision	recall	f1-score	support
Foxy	0.91	0.92	0.92	117
Jerry	0.81	0.96	0.88	117
Mickey	0.94	0.87	0.90	117
Milton	0.95	1.00	0.97	117
Tom	0.95	0.78	0.85	117
accuracy			0.91	702
macro avg	0.91	0.91	0.91	585
weighted avg	0.91	0.91	0.91	585

Batch Size 6 with Weight Decay 0.0001: We observed worse overall accuracy, recall and precision for the Mickey class.

Table 6: Classification report for batch size 6, weight decay 0.0001

	precision	recall	f1-score	support
Foxy	0.79	0.95	0.86	117
Jerry	0.80	0.95	0.87	117
Mickey	0.94	0.70	0.80	117
Milton	0.95	1.00	0.97	117
Tom	0.94	0.76	0.84	117
accuracy			0.87	702
macro avg	0.88	0.87	0.87	585
weighted avg	0.88	0.87	0.87	585

Batch Size 4 with Weight Decay 0.01: This setting provided no improvement for the Mickey class, overall accuracy, nor did it lower the validation learning loss.

Table 7: Classification report for batch size 4, weight decay 0.01

	precision	recall	f1-score	support
Foxy	0.91	0.91	0.91	117
Jerry	0.81	0.96	0.88	117
Mickey	0.93	0.87	0.90	117
Milton	0.95	1.00	0.97	117
Tom	0.95	0.77	0.85	117
accuracy			0.90	585
macro avg	0.91	0.90	0.90	585
weighted avg	0.91	0.90	0.90	585

Batch Size 6 with Weight Decay 0.01: We observed a loss in the recall and precision balance seen in the previous setting for the Mickey class and a lowered overall accuracy.

Table 8: Classification report for batch size 6, weight decay 0.01

	precision	recall	f1-score	support
Foxy	0.78	0.95	0.86	117
Jerry	0.78	0.93	0.85	117
Mickey	0.90	0.71	0.79	117
Milton	0.97	0.97	0.97	117
Tom	0.92	0.74	0.82	117
accuracy			0.86	585
macro avg	0.87	0.86	0.86	585
weighted avg	0.87	0.86	0.86	585

References

Walt disney productions, ltd. v. pathe exchange, inc. and the van beuren corporation (equity t-87), 1932. National Archives Identifier: 333398767.

Beck, J. and Friedwald, W. *Looney Tunes and Merrie Melodies: A Complete Illustrated Guide to the Warner Bros. Cartoons*. Henry Holt and Co., 1989. ISBN 0-8050-0894-2.

Bellos, D. and Montagu, A. *Who Owns This Sentence?* W. W. Norton & Company, 2024. ISBN 9781324073710.

Ginsburg, J. C. and Budiardjo, L. A. Authors and machines. *Berkeley Technology Law Journal*, 34: 343, 2019. URL https://scholarship.law.columbia.edu/faculty_scholarship/2323.

Google. How contentid works. Google Support, 2023. URL https://support.google.com/youtube/answer/2797370?hl=en. Accessed: 2023-04-13.

Google Creative Lab. Quick, draw! dataset. https://github.com/googlecreativelab/quickdraw-dataset, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *arXiv*, abs/1512.03385v1, 2015. URL https://arxiv.org/abs/1512.03385.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations (ICLR)*, 2020.

IMDB. Rudolf ising, n.d. URL https://www.imdb.com/name/nm0411208/?ref_=nm_ov_bio_lk.

Kim, D., Heo, S., Kang, J., Kang, H., and Lee, S. A photo identification framework to prevent copyright infringement with manipulations. *Applied Sciences*, 11 (19):9194, 2021. doi: 10.3390/app11199194. URL https://doi.org/10.3390/app11199194.

Lee, K., Cooper, A. F., and Grimmelmann, J. Talkin' 'bout ai generation: Copyright and the generative-ai supply chain. *Journal of the Copyright Society*, July 2023. forthcoming.

Li, M., Yumer, E., and Ramanan, D. Budgeted training: Rethinking deep neural network training under resource constraints. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Scheffler, S., Tromer, E., and Varia, M. Formalizing human ingenuity: A quantitative framework for copyright law's substantial similarity. In *Proceedings of the 2022 Symposium on Computer Science and Law*, pp. 37–49, 2022.

Shermin, T., Teng, S. W., Murshed, M., Lu, G., Sohel, F., and Paul, M. Enhanced transfer learning with imagenet trained classification layer. *arXiv*, September 2019.

United States District Court for the Northern District of California. Procedures and tentative rulings for may 8, 2024 hearing. United States District Court Northern District of California Document, 2024. URL https://fingfx.thomsonreuters.com/gfx/legaldocs/egpbazjyyvq/STABILITY%20AI%20COPYRIGHT%20LAWSUIT%20tentative.pdf. Case No. 23-cv-00201-WHO.

U.S. Court of Appeals for the Ninth Circuit. Sid & marty krofft television prods. v. mcdonald's corp. *Federal Reporter, 2nd Series*, 562:1157, 1164, 1977.

U.S. Court of Appeals for the Ninth Circuit. Fisher v. dees. *Federal Reporter, 2nd Series*, 794:432, 434 n.2, 1986.

U.S. Court of Appeals for the Ninth Circuit. Olson v. national broadcasting co., inc. *Federal Reporter, 2nd Series*, 855:1446, 1449, 1988.

U.S. Court of Appeals for the Ninth Circuit. Apple computer, inc. v. microsoft corp. *Federal Reporter, 3rd Series*, 35:1442, 1994.

U.S. Court of Appeals for the Ninth Circuit. Newton v. diamond. *Federal Reporter, 3rd Series*, 388:1189, 1193, 2004.

U.S. Court of Appeals for the Ninth Circuit. Antonick v. elec. arts, inc. *Federal Reporter, 3rd Series*, 841:1066, 2016.

U.S. Court of Appeals for the Ninth Circuit. Rentmeester v. nike, inc. *Federal Reporter, 3rd Series*, 883:1118, 2018.

U.S. Court of Appeals for the Second Circuit. Arnstein v. porter. *Federal Reporter, 2nd Series*, 154:464, 1946.

U.S. Court of Appeals for the Second Circuit. Peter pan fabrics, inc. v. martin weiner corp. *Federal Reporter, 2nd Series*, 274:487, 489, 1960.

Walt Disney Productions, Ltd. v. Pathe Exchange, Inc. and the Van Beuren Corporation. Exhibit, 1932. URL https://www.archives.gov/riverside/highlights/walt-disney-mickey-mouse. National Archives Identifier: 333398767.