Project Hentai AI: wAiFu
Reviewing Anime Thighs with Deep Learning

1st hentai-ai
Project Hentai AI
https://www.hentai-ai.org

***Abstract*—For too many years have the world of Artificial Intelligence (AI) and the world of Hentai been separate ecosystems. Project Hentai AI aims to bring AI into the sphere of Hentai, Ecchi and Lewds. In this paper, we propose a Witty Artificial Intelligence Framework Utilization (wAiFu). This framework is built for processing and labeling data, as well as training machine learning models to classify images of lewd anime/manga and hentai based on subjective user rating. As a proof of concept, this framework is applied to images of lewd anime thighs labeled using a boolean method. A dataset of 1000 images is collected, processed and labeled before being loaded into a fastai implementation of a Convolutional Neural Network (CNN) designed for Computer Vision. The retraining of a resnet34 model for 20 epochs using labels from three different users resulted in an accuracy of 70%, 78% and 97%. Furthermore, a couple of limitations were identified, most importantly that the small size of the dataset could cause the model to overfit. As mitigation, the data was augmented using batch transforms in fastai. Future work in Project Hentai AI will focus extra on upscaling the data collection phase.**

***Index Terms*—deep learning, DL, machine learning, ML, artificial intelligence, AI, thighs, thighdeology, thicc, lewd, ecchi, hentai**

I. Introduction

It all began when a friend started reviewing anime thighs sent their way. The reviews were simply approved or disapproved, but the surprisingly low amount of approved images sparked the idea of a machine learning model capable of learning an individual's taste in anime thighs. By feeding a model images with rating labels, it could be able to learn an individual's subjective taste. The framework of wAiFu is not limited to lewd anime thighs, but can very easily be extended to other features e.g., tits, ass, abs, middriffs and armpits. The code of all tools in Project Hentai AI is open source and can be found at https://git.hentai-ai.org.

II. Background

A. Artificial Intelligence

Artificial Intelligence (AI) is an umbrella term for the area in computer science aiming to artificially create an intelligent software using statistics and algorithms. There is an important distinction here between Intelligence and Consciousness. An AI which can calculate the best move in chess could be considered intelligent, but does not necessary have a consciousness (a notion of self). The simplest forms of AI are the Non-Playable Characters (NPCs) and bots of video games. The main goal of these AI is to emulate human behavior in order to create an illusion of intelligence and/or consciousness.

Machine Learning (ML) is a subset of AI which is best described by Tom M. Mitchell [5]:

> *"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E"*

This means that the algorithm gain experience by training on a task and that this training can then be measured. And the more the algorithm train and gain experience, the better it performs on the task. These tasks are usually classification tasks in ML (e.g., classify email as spam or separating images of cats from images of dogs).

While ML needs to perform the feature extraction manually from the input before classification, *Deep Learning* (DL) neural networks automatically extracts the features as a part of the classification [4]. DL also uses backpropagation algorithms to adjust the parameters of hidden layers (between the input and output layers) during training. Due to its feature extraction, DL can work on both structured and unstructured data as input, and this in turn has made DL efficient in object detection and speech recognition, both of which are classification problems (e.g., does the *sound* match any known *word*).

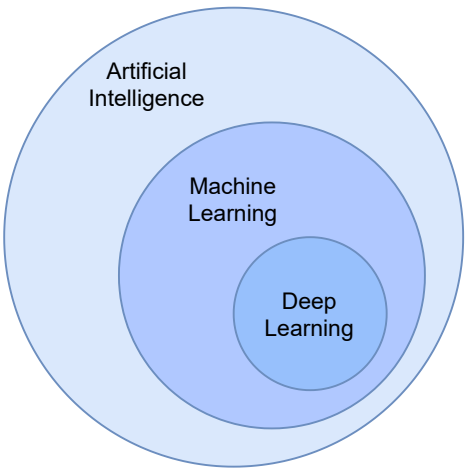

Fig. 1. Relation between Artificial Intelligence, Machine Learning and Deep Learning.

Machine Learning and Deep Learning falls under the discipline of Artificial Intelligence in computer science, visually presented in Figure 1.

B. Hentai and Thighdeology

For the purpose of this and future studies in Project Hentai AI, the data in the datasets are categorised in three definitions: *Hentai*, *Ecchi* and *Lewd*. In its simplest definition, Hentai can be described as anime and manga pornography. Ecchi on the other hand, when used as an adjective, translates to "sexy", "dirty" or "naughty", and has been used to describe anime and manga with *sexual overtones* (playful sexuality or softcore). Lewd in these studies is defined as *sexual undertones*. A detailed differentiation between the three categories is planned for a separate study. Project Hentai AI will include ecchi and lewds, even though the name of the project uses the term hentai.

Thighdeology is the worship of thick anime thighs which has its Mecca on the Thighdeology subreddit [6]. The top two rules on the subreddit are: (1) All images must be thigh-focused and (2) No Pictures of Sex (Nudity is allowed). The second rule is a clear demonstration of the distinction between hentai and ecchi described above. The dataset used for wAiFu is images of lewd anime thighs in accordance with these two rules. The epigraph which crowns the website says it all:

> *"Blessed is the man that walketh not in the counsel of the ungodly, nor standeth in the way of sinners, nor sitteth in the seat of the scornful. But his delight is in the law of the THICC anime thighs."*

III. Method

wAiFu stands for Witty Artificial Intelligence Framework Utilization, and its goal is to standardize the process of creating a subjectively labeled dataset for machine learning. This means that a single set of images can be used as separate datasets depending on the subjective labeling. A script homogenized the images (filename and file extensions). An application was developed for cropping the images to isolate the area of interest as much as possible. Finally an application was developed to label the images using a separate file for mapping each filename to its subjective labeling.

IV. Design

The following section describes the design of wAiFu in its separate components in detail: the data collection, the data preparation, the data labeling and finally the machine learning API.

A. Data Collection

A dataset of lewd anime thighs was manually collected from six separate sources:

- Discord Server: All Things Hentai
- Discord Server: Hanako's Hideout[1]
- Discord Server: hanime.tv Community
- Discord Server: NCE: The NEKOPARA Community
- Subreddit: Thighdeology
- Private Donations

[1]formerly known as r/Hentai Group prior to 13th April 2021

After collection, the data was manually screened for (A) presence of thighs (B) image quality and (C) image *cropability*. The presence of thighs simply implies that the image in question contains a section of the lower body of a humanoid character. The vast majority of the characters depicted in the images collected were of the feminine nature, although this was most likely due to the skewed ratio of feminine/masculine thighs from the sources themselves and not due to any discrimination during the manual collecting. This is included within future work in Section VII-B. Image quality refers to the resolution of the picture. When finding duplicates, the one with higher resolution was kept. Some images where included in the dataset due to its contents, even if the quality of the resolution was below average. Image cropability refers to the composition of the picture. Since the focus of the first dataset in wAiFu is *thighs*, it is preferred to isolate the thighs from other factors in the image which could influence the labeling, such as: faces, tits and other eye-catching details (some of the cropped images in the dataset does contain the ass region due to non-perfect but acceptable levels of cropability).

B. Data Preparation

The data preparation in this project consist of three stages after being collected:

1) Converting
2) Renaming
3) Cropping

In order to get a uniform dataset the images collected were converted from JPG/JPEG to PNG. The naming convention was arbitrarily decided to be structured as **thighs-id.png** where **id** is an increasing nonce (number only used once) padded with four zeroes (e.g., **thighs-0001.png**).

The images were then cropped to contain as little as possible apart from the topic at hand (thighs). This was done with the intention of focusing both the manual labeling process as well as the machine learning training on the thighs. If the character on the image would have a certain unrelated feature this could potentially influence the user when labeling the dataset, and later might be picked up during the learning and thus distorting the focus on the subject matter for this study.

After cropping the original non-cropped images are kept with their original name, while the newly cropped images get an appended notation of having undergone the procedure (e.g., **thighs-0001-crop.png**). The cropped images were placed in a separate directory from the original images. By keeping both datasets, this study provides the possibility of utilizing the non-cropped images for future work. The data preparation implementation is detailed in Section V-A.

C. Data Labeling

The labeling of datasets in wAiFu is categorised in two different methods:

- Boolean labeling
- Scale labeling

Fig. 2. The protocol of reviewing thighs using boolean labeling

The *Boolean labeling* consist of two disjunctive values (e.g., True/False, Yes/No, Approved/Disapproved, 1/0) which is the closest to the responses gotten previously when brokering pictures of anime thighs manually. An image would be sent and an Approved/Disapproved would be received in return. A diagram example is seen in Figure 2.

The *Scale labeling* ranks the images on a scale (e.g., 0-10, 1-5, A-F). This could be considered to be an extension of Boolean labeling (which would be seen as a scale of 0-1) by adding float values in between. The scope of this study will cover boolean labeling only, but scale labeling is included in future work.

The data labeling implementation is detailed in Section V-B

D. fastai

The AI implementation is using fastai, a deep learning library providing machine learning practitioners with high-level components creating state-of-the-art results in standard deep learning domains [3]. For the purpose of boolean labeling in this project, a single-label classification structure is implemented using various building blocks. The pictures and their labels are loaded into a *DataLoaders* object. This object is responsible for maching labels with images, applying item transforms (transforms applied to each image individually) and batch transforms (transforms applied to each batch during training). It is also responsible of splitting the dataset into various sets: *training, validation* and *testing* (see Figure 3). The training set is used to train a given model, which sees and learns from this data. The validation set is used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. Unlike the training set, the model only occasionally sees this data but never learns from it. The testing set is used to provide an unbiased evaluation of a *final model fit* of the training dataset.

The DataLoaders object is then combined with a model and a metric to create a *Learner* object. The model can be pre-trained, which means that some object and shape recognition

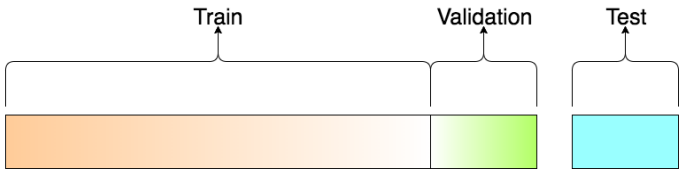

Fig. 3. Visualization of the three sets of data

can be used as a foundation to train a model for a more specific computer vision problem. This method is called *transfer learning*. The Learner object has a bunch of methods including: `fine_tune`, `validate` and `export`. The `fine_tune` method first freezes all layers except the last one for one cycle (a "prequel" epoch), and then unfreezes all layers before running the epochs. This process of freezing and unfreezing layers in the Convolutional Neural Network improves the performance of transfer learning. So using `fine_tune(2)` would first run a cycle only adjusting the last layer, then run 2 epochs adjusting all layers. The `validate` method is simply running predictions on a set of data and comparing the predictions to the actual labels. This is done after the training to sample the accuracy of the model on a testing set. The `export` method saves the trained model to a file for future use.

V. Implementation

A. Data Preparation

The following section goes through the implementation of homogenizing the dataset, including renaming, changing extensions and cropping the images.

1) Convert and Rename: The homogenization of the dataset is done with mainly two functions: `RenameFiles()` and `ConvertFilesToPng()`. All image files in the dataset is renamed to the naming convention detailed in Section IV-B unless already matching the pattern. The images are also converted to png-files if they are not.

2) Cropping Images: The application for efficiently cropping the images manually was built ontop of a zooming-application [2] which utilizes tiling for increased performance. The frame border of the application window was set to a 1:1 aspect ratio with desired dimensions and could then easily be used to crop every image from a specified input directory, and put the cropped images in a separate (or in the same) destination directory. Figure 4 and Figure 5 shows before and after cropping with the application. The code is open source and can be found at: https://git.hentai-ai.org/?p=hentai-cropper.git/.git

B. Label App: Hentai Tinder

The name of the label application is "Hentai Tinder" (see Figure 6). It is written in Python using the Tkinter library, a Python binding to the Tk GUI toolkit [1]. The application loads the images in batches and presents each image to the user one-by-one. The GUI consists of four buttons, inspired by the original Tinder application: *Smash, Pass, Go Back* and

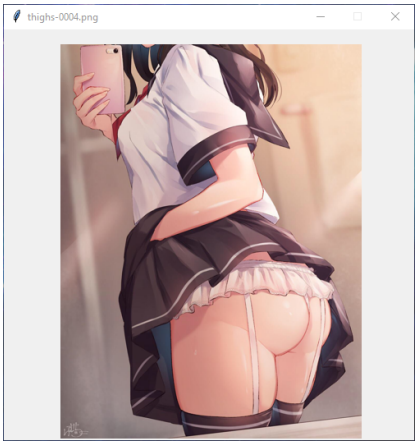

Fig. 4. Image before cropping with application

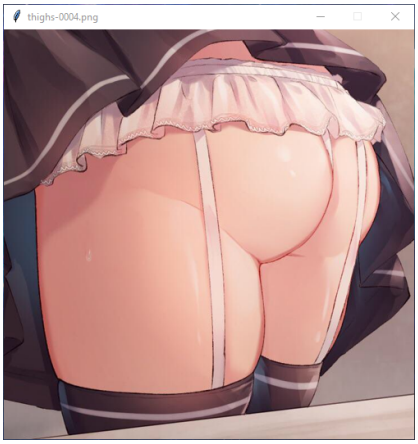

Fig. 5. Image after cropping with application

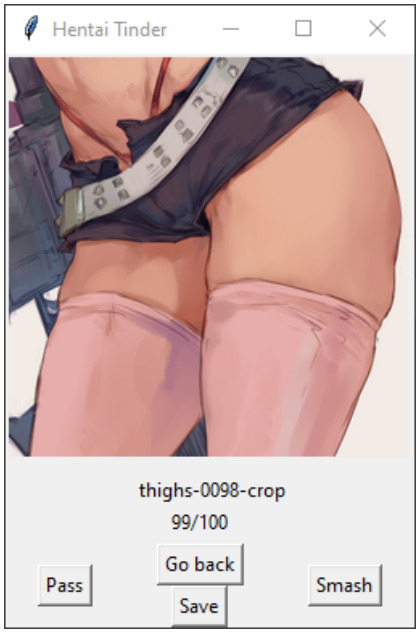

Fig. 6. Hentai Tinder

```
fname,label
thighs-0000-crop.png,False
thighs-0001-crop.png,False
thighs-0002-crop.png,True
thighs-0003-crop.png,False
thighs-0004-crop.png,False
```

Fig. 7. Structure of output file using comma separated values

Save. *Smash* will label the image as "True" and *Pass* will label it as "False". These are internal boolean values which represents if the user liked the image or not. *Go Back* was added in order for users to change their mind about previous images. The *Save* button simply writes the current results to the output file. The output of the Hentai Tinder application is a csv file which can be used in fastai to create a dataloader with all the images including their labels, see Figure 7.

The code is open source and can be found at: https://git.hentai-ai.org/?p=hentai-tinder.git/.git

C. Deep Learning with fastai

The deep learning framework (fastai) was implemented using interactive python notebooks running on Google Colab[2] connected to Google Drive[3] for storing csv-files, dataset and trained models. The dataset is loaded using the pandas library from the csv-file generated by Hentai Tinder. Then the dataset is split into a 8:1:1 set of training, validation and testing. Transforms are specified for the training and validation set in the dataloader, which is then added to the learner. The learner uses the dataloader and a downloaded resnet34 for the training. The model is trained for 20 epochs and saved to a file. The testing set is then used to get the final accuracy measurements. The notebook is open source and can be found at: https://git.hentai-ai.org/?p=waifu-notebook.git/.git

[2]https://colab.research.google.com
[3]https://drive.google.com

VI. Results

A. Justifying Additional Transforms

One of the main observations when training on such a small dataset was the tendency to overfitting. There are two types of transformations applied to the dataset before training: `item_tfms` and `batch_tfms`. The item_tfms for this implementation is using *RandomResizedCrop* which will crop every image randomly to 224x244 with a minimum scaling of 0.75. The batch_tfms is applying many more tranformations to images in batches between each epoch. These transformations include: zooming, flipping, rotating and changing the brightness. Figure 8 shows how *only* item_tfms transform the dataset. Figure 9 shows how batch_tfms additionally transforms the dataset further. Figure 10 shows the batch_tfms's effect on error_rate, train_loss and valid_loss. The loss for both the training and validation set should be as low as possible. A high loss indicates a failure to generalize the learning. We

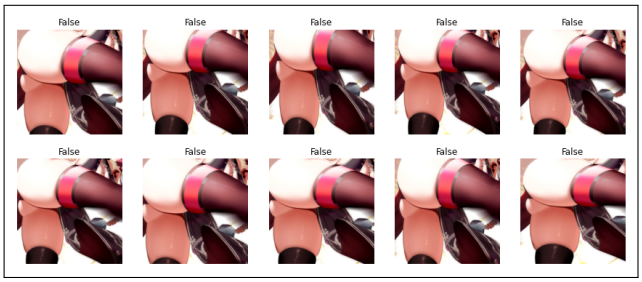

Fig. 8. Training image without batch transforms

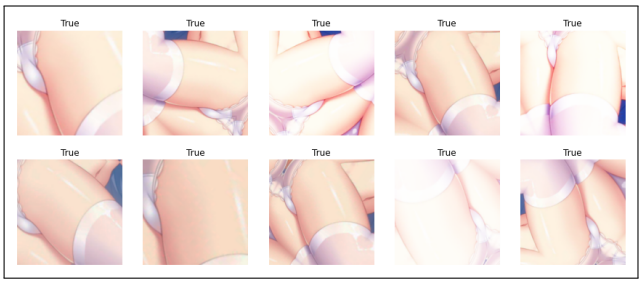

Fig. 9. Training image with batch transforms

see in the bottom most graph in Figure 10 that the validation loss is dramatically decreased using batch transforms.

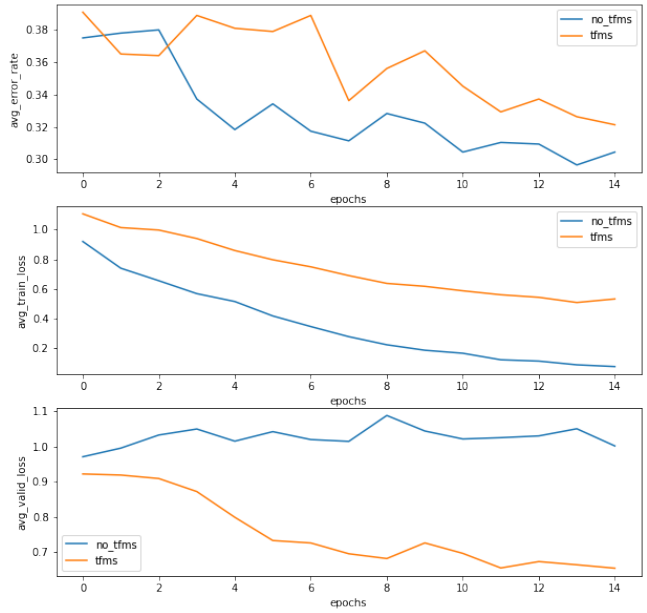

Fig. 10. Comparing with and without batch transforms on error_rate, train_loss and valid_loss

B. Error Rate of Thighs

The dataset containing 1000 images was labled using Hentai Tinder (Section V-B) by three individual people: User A, User B and User C. A table of the training result after 20 epochs for each user can be seen in Table I. The three users had varying rates of approval on the dataset with user C liking almost half of the dataset. With the error rate sometimes being close to the rate of approval, a sanity check with a confusion matrix showed that the model did not just predict true/false on the whole dataset or started overfitting. In Table II we show the true/false positive/negative results on the testing set for each user. Furthermore we show the accuracy on the testing set using:

TABLE I
USER TRAINING

User	Approved	train loss	valid loss	valid err
A	33.00%	0.42	0.64	32%
B	13.22%	0.33	0.53	20%
C	49.30%	0.47	0.54	26%

TABLE II
USER TESTING

User	TP	FP	TN	FN	Accuracy
A	11	13	67	9	78%
B	1	3	96	0	97%
C	44	8	26	22	70%

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

VII. DISCUSSION

A. Limitations

The size of the lewd anime thighs dataset is only 1000 images. This leads to overfitting during training which can be mitigated slightly by applying transformations. The small dataset is due to the time-consuming task of manually cropping and labeling the dataset. Since the model is trying to learn an individual's taste, the individual must label the full dataset.

B. Future Work

In order to increase the size of the dataset and thereby obtaining a more robust accuracy from the machine learning model, future research in Project Hentai AI will spend some more focus on automating the collection, transformation and labeling of data.

In this study, only boolean labeling was considered when reviewing lewd anime thighs. But even in the world of Hentai, thighs are more often than not in a gray-zone as opposed to black or white. Future work would be to extend the labeling application (*Hentai Tinder*) to have a mode or a version capable of using rate labeling on a scale. This could be as easy as presenting the user with a 5-star system, similar to reviewing restaurants or hotels, where each image gets rated from 1-5.

As metioned in Section IV-A, the dataset mainly contained lewd feminine thighs. One area of future work would be to investigate the masculine/feminine feature ratio effect on the model accuracy robustness. In other words using other/larger datasets.

VIII. Conclusion

In Project Hentai AI: wAiFu (Witty Artificial Intelligence Framework Utilization) we established a framework for processing and labeling data using our own newly developed tools. We re-trained a Convolutional Neural Network using fastai to classify images of lewd anime thighs based on subjective ratings from three individual users with an accuracy of 70%, 78% and 97%. Even though batch transforms where applied to mitigate overfitting, we believe that the dataset could still be too small. The size of the dataset is impacted by the pre-processing overhead (cropping) of the general dataset images, as well as the manual labeling for each new user.

Acknowledgement

We would like to thank Kittey for coming up with the name of the project: *wAiFu*. We would also like to thank Hood Classic#0148 for coming up with the name of the labeling app: *Hentai Tinder*. Finally we thank the three anonymous users in this study.

References

[1] Python Documentation. Tkinter. docs.python.org/3/library/tkinter.html, March 2022.

[2] foobar167. Github: zoom_advanced.py. https://github.com/foobar167/junkyard/blob/master/zoom_advanced.py, March 2022.

[3] Jeremy Howard and Sylvain Gugger. Fastai: A layered API for deep learning. *Inf.*, 11(2):108, 2020.

[4] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nat.*, 521(7553):436–444, 2015.

[5] Tom M Mitchell et al. Machine learning. 1997.

[6] Reddit. Thighdeology subreddit. www.reddit.com/r/thighdeology, March 2022.