# Forecast Views Through Video Cover Images —A Classification Problem Using VGG and ResNet from Machine Learning

**Wenyu Xu Qiaowei Li**

*Abstract—Although people are aware that a good thumbnail stands out and draws viewers' attention intuitively, few studies have looked at the relationship between thumbnails and views from a machine learning perspective. Many articles just qualitatively analyze what kind of thumbnail is attractive, giving some design tips. Our project analyzes problems quantitatively, predicting a view range according to an input picture. We use both VGG and ResNet from machine learning to train a model for three datasets. Based on the model we build, our project proves that video views are related to its cover image and verifies some qualitative studies mentioned above. Our model helps the video producer to check whether his video cover is attractive enough in a targeted manner. Further developing, it might help us study human visual aesthetics.*

## 1 INTRODUCTION

On a video platform, like YouTube, people are often attracted toward a cover picture of a video, which contains some information and determines whether the web surfer will click to view it. On the other hand, from the video creators' side, a good thumbnail stands out and can draw viewers' attention. From the human visual aesthetic aspect, it seems feasible that a video's hits can be somehow related to its cover.

In fact, many articles have concluded some cover image strategies that could increase the appeal of a video. For example, according to the study called *How do you get more people to click on your video? – 12 tips for optimizing YouTube thumbnails*[1], images with clear emotional expression, a close-up shot of someone's face, or human eye contact can have a better performance on click rates. Using bright color as the background and putting words in it are also good strategies. Another plausible feature is whether there is an important element on the cover image's right side corners. The video length and watch later label in the same position might block important information the author wants to show, thus reducing the attractiveness. There are many more such thumbnail optimization strategies people discuss online.

However, although we know that there is a certain relationship between video cover and views, many articles qualitatively analyze what kind of thumbnail is attractive, and there is basically no case-by-case study. Our project analyzes problems quantitatively, predicting the stable click rate range according to the input picture. It helps the video producer to test whether his video cover is attractive enough in a targeted manner. Also, further developing, it might help us study human visual aesthetics.

Through web scraping, we obtained 11,813 thumbnails and corresponding views data from Bilibili (a Chinese video media platform). We chose a particular type of video—vlog—to study since cover images of the same category are more comparative. Using three datasets, we trained the model using both VGG and ResNet techniques for each. It turned out that for the two imbalanced datasets, the model performance is not that good. But for the relatively balanced one, the model accuracy performs better, exceeding the baseline around 5-10%. So it is reasonable that the cover image is related to the view numbers. We also apply the model to real cover image design and verify some image design tips mentioned above.

## 2 DATASET

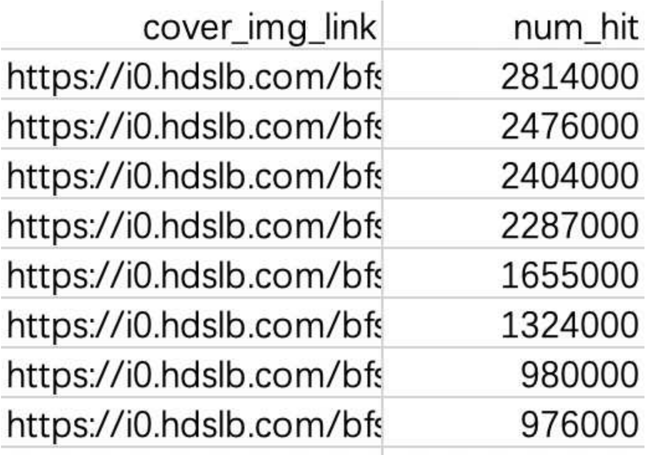
In this section, we will discuss the datasets collection, pre-processing and our final datasets.

### 2.1 Datasets Collection

To forecast views through video cover images, the two attributes needed are the video cover image and the corresponding number of views.

At first a clean and suitable dataset on Kaggle called *Trending Youtube Video Statistics* was found. It has 16 attributes, including the video thumbnail link and its view number. One of the sub-datasets is the statistics of US region videos. It has 40,949 entries in total, but the trending dates are all in 2017 (three years ago). Considering the potential change in the cover image trend, to eliminate the plausible influence of this time factor, we decide to use web crawling to get first hand dataset that has a closer time range.

The first step to start web crawling is to choose a suitable video platform to conduct our research. After investigation on content structure and characteristics, the Chinese video website www.bilibili.com is chosen, and the detailed



| cover_img_link | num_hit |
|---|---|
| https://i0.hdslb.com/bfs | 2814000 |
| https://i0.hdslb.com/bfs | 2476000 |
| https://i0.hdslb.com/bfs | 2404000 |
| https://i0.hdslb.com/bfs | 2287000 |
| https://i0.hdslb.com/bfs | 1655000 |
| https://i0.hdslb.com/bfs | 1324000 |
| https://i0.hdslb.com/bfs | 980000 |
| https://i0.hdslb.com/bfs | 976000 |

Fig. 1. A clip of the csv file.The csv file contains mainly two columns: cover image link and the number of hits.

reasons are as follows. First, Bilibili is a hub for Chinese video creators. Compared to other large Chinese video platforms, Bilibili has a larger and clearer collection of videos uploaded by different people. Second, there hasn't been a lot of research into the nature of Bilibili's video cover images, since current studies usually focus on YouTube.

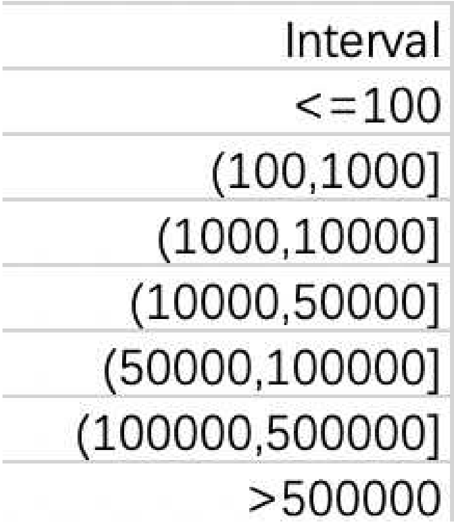
Considering the fact that different video categories have different cover image trends, we choose to target one specific category, because images of the same category are more comparative. Since vlog is now in fashion and more and more people are using it to record their lives, our dataset focuses on the cover image of the "vlog" category (video blogs).

To get the cover image link and view number of the videos on Bilibili, the method web scraping is used. We search for the videos that have a "vlog" tag, collect the html element of the webpage, and scrape the cover image link and views using Beautifulsoup4 , which is a Python package for parsing HTML and XML documents, and write all the information in a csv file, as shown in Figure 1.

### 2.2 Datasets Preprocessing

Our question is that given a cover image, how we can predict its views. We define this question as a classification problem instead of regression, since some video views are not stable, especially for newly uploaded videos; using such data to build a regression is not accurate, and using the inaccurate model to predict an exact view number for the input does not make sense. So instead, we choose to build a classification model of Machine Learning to give a range of views for a certain input.

According to bilibili's mechanism, there are different rewards for different video view levels. For example, if one's video gets more than one thousand views, his or her account will get a special icon; if it gets more than ten thousand views, his or her account will get a small medal. Based on this mechanism, along with the distribution histogram of the view number we have ,it seems reasonable for us to divide the dataset into seven categories, as shown in figure 2 below. Image data downloading and class folder separation is done within the python file for web scraping. To ensure all downloaded images are valid and complete, another python function is written to check if the image is downloaded properly.



| Interval |
|---|
| <=100 |
| (100,1000] |
| (1000,10000] |
| (10000,50000] |
| (50000,100000] |
| (100000,500000] |
| >500000 |

Fig. 2. The dataset is divided into seven categories.

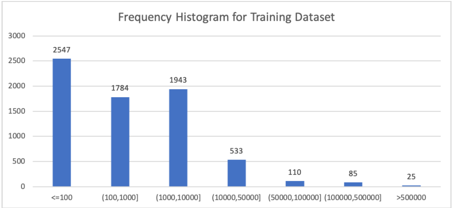


Fig. 3. The distribution of the dataset is very imbalanced.

### 2.3 Result Dataset

The first training set obtained has about seven thousand entries, however the distribution of the 7 classes is very imbalanced, as shown in figure3. The first three classes contains many more examples than the rest classes. Problems might appear with an imbalanced dataset, since the classifiers tend to ignore small classes while concentrating on classifying the large ones accurately[2]. Then the predictive accuracy used to evaluate the performance of a classifier might not be appropriate due to the imbalance dataset.

To eliminate this problem, we choose to act on the data. We first use the method of oversampling: to find more data for the minor classes. However, due to the fact that there are many many more videos that has a small number of views than those that have a high number of views and considering the factor of the search algorithm of the video platform, we are not able to add enough data to make the classes equally distributed, so another sampling method is conducted: undersampling. Undersampling is to remove some data points from the majority classes[reference:https://towardsdatascience.com/guide-to-classification-on-imbalanced-datasets-d6653aa5fa23], and here random undersampling is used, which is to randomly

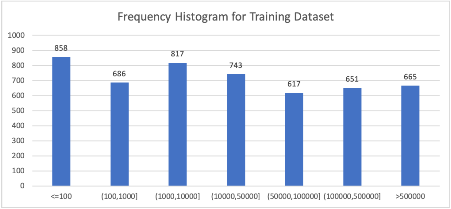


Fig. 4. A balanced but smaller dataset.

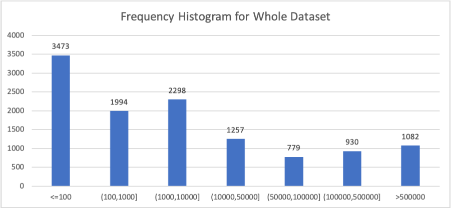


Fig. 5. The entire dataset we have collected.

select examples from the majority classes and delete them from the training set.

After the above two steps, we get our second dataset, which is a balanced but smaller dataset with 5 thousand entries in the training set, as shown in figure4.

To be more considerate, a third, also the last dataset is formed. In this last dataset, all data once found is included, all data before undersampling them to a balanced dataset. We keep this dataset to prevent the plausible limitations ofundersampling, because it is unknown to what extent the removed data is useful or important in determining the decision boundary between the classes, so the third dataset is used to prevent important information being deleted.

This final dataset contains around 11813 entries, and we split it by the ratio of 6:2:2 for the training set, validation set, and test set since we have enough data to do proper held-out test data[3].

## 3 EXPLANATION OF THE METHOD USE

In this section, we will discuss the datasets collection, pre-processing and our final datasets.

The machine learning method we choose for solving our image classification problem is convolutional neural network(CNN) and Transfer Learning.
To process images as input, CNN is needed. Transfer learning here is a machine learning technique used to enhance the performance of our classification model. What transfer learning does is to reuse a pre-trained model for another related task, make certain modifications so that the model serves our own purpose. The reason for using Transfer learning in our project is that, in practice, compared to the open dataset online like ImageNet that has over 10 million images, the dataset we collected by our self is still too small, so using others‘ pre-trained models and weights as our own initialization will optimize the performance of our model, as we do not have such an exceptionally large dataset.

The pretrained models we use is VGG16 and ResNet50, two famous public models for transfer learning.

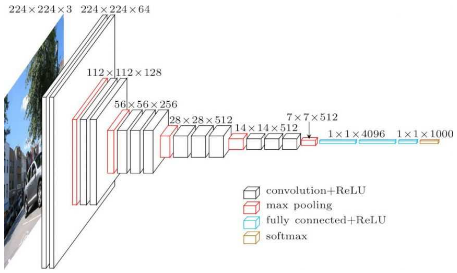


Fig. 6. The Architecture of VGG16.

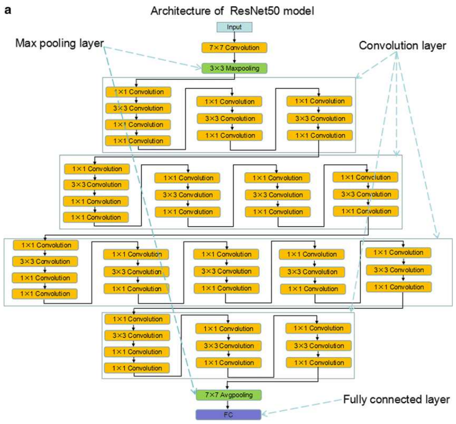


Fig. 7. Architecture of ResNet50.

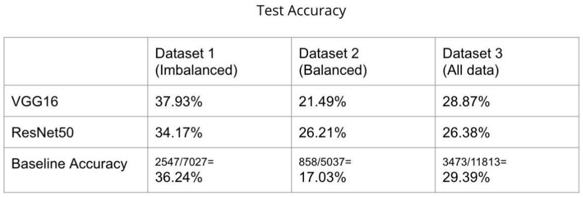
VGG16 is a convolutional neural network model that has achieved 92.7% top-5 test accuracy in ImageNet, a dataset of over 14 million images belonging to 1000 classes.The input to it must be RGB image of size 224 x 224 . The image passes through a stack of convolutional layers, where the filters were used with a very small receptive field: 3×3 [4].

ResNet50 is a deep residual learning framework, a variant of ResNet model which has 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer. On the ImageNet dataset residual nets are evaluated with a depth of up to 152 layers, which is 8 times deeper than VGG nets but still having lower complexity.[5]

What we do is to freeze the front layers of the model, use them to get a feature vector representation of the input image as our initialization and change the last layer to logistic regression that suits our classification problem. The logistic regression is a fast algorithm which outputs class probabilities with high interpretability and popularity. It applies “linear regression” to a non-linear transformation of Y. Since we have a relatively large dataset of images, this fast algorithm can be our primary choice.

## 4 RESULTS AND DISCUSSION

In this section, we will discuss the datasets collection, pre-processing and our final datasets.



Test Accuracy

| | Dataset 1<br>(Imbalanced) | Dataset 2<br>(Balanced) | Dataset 3<br>(All data) |
|---|---|---|---|
| VGG16 | 37.93% | 21.49% | 28.87% |
| ResNet50 | 34.17% | 26.21% | 26.38% |
| Baseline Accuracy | 2547/7027=<br>36.24% | 858/5037=<br>17.03% | 3473/11813=<br>29.39% |

Fig. 8. summary of test accuracy of the 6 models.

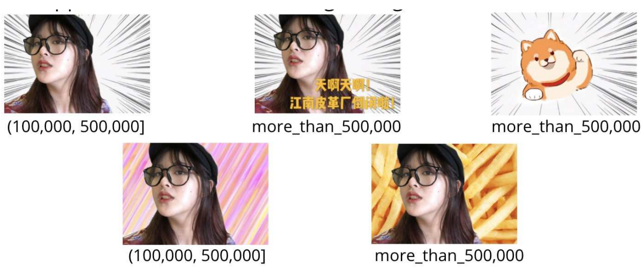


Fig. 9. Holding other variables the same, we change one variable at a time..

Here's the result. From Figure 3, for the imbalanced dataset 1 and 3, the performance is not that satisfactory compared with the baseline accuracy. But for the balanced dataset 2, the model performs better, exceeding the baseline accuracy around 5-10%. By data, our project proves the argument that video views are related to its cover image.

Based on the test outcome, the best model among all is ResNet50 trained with Dataset 2, which exceeds the baseline performance around 10%. Therefore, we choose this model to predict the views of other images. To see whether the thumbnail design tips, as mentioned in the introduction part, are right, we design several sample cover images to test. Holding other variables the same, we change one variable at a time, such as background color, human face and words . It turns out that some thumbnail design tips are reasonable. The model we run gives us some predictions about the input images, as shown in the figure. We can see that using a bright color as the background and putting words on the image are useful strategies. On the other hand, according to the tips, thumbnails with a clear human face are more likely to get more views. However, Image and Image show the opposite, which implies our model can be improved.

## 5 FUTURE WORK

In the short term, if we had more time, there are several steps we envision to take. First, we want to build a website connected to our model so that people can more easily test the attractiveness of their video covers. And therefore, the model we produce can be widely spread and used. Second, we want to refine the model to improve its performance. We will try to retrain some layers of CNN so that our model can be more accurate. Third, though we've already used the confusion matrix to test the accuracy, we may want to apply ROC for multiclass classification to get another direct visualization.

In the long term, trying video platforms in more countries may help us study the cultural pattern in different parts of the world. So we can adjust the prediction model considering the social and cultural factors. Also, trying different algorithms for the last layer in CNN or retraining the front-end layers might be some ways to improve the model. There's lots of knowledge in thumbnail selection, video highlighting and summarization, and computational aesthetic for us to study, so that we can understand the problem more deeply. Our study on image attraction for people may have some contributions to human visual aesthetic by further improving the model.

## References

[1] *"How do you get more people to click on your video? – 12 tips for optimizing YouTube thumbnails"* https://zhuanlan.zhihu.com/p/67295505

[2] Matthew Stewart *Guide to Classification on Imbalanced Datasets.* https://towardsdatascience.com/guide-to-classification-on-imbalanced-datasets-d6653aa5fa23

[3] https://stackoverflow.com/questions/13610074/is-there-a-rule-of-thumb-for-how-to-divide-a-dataset-into-training-and-validatio: :text=Split%20your%20data%20into%20training,performanc

[4] https://neurohive.io/en/popular-networks/vgg16/

[5] https://www.kaggle.com/keras/resnet50

[6] https://iq.opengenus.org/resnet50-architecture/