Large Scale Empirical Study of Image Stylization Repositories hosted on GitHub and Stack Overflow

Ikram Hussain
ikramhussain@cmail.carleton.com
Computer Science, Carleton University
Ottawa, Ontario, Canada

Projna Saha
projnasaha@cmail.carleton.ca
Computer Science, Carleton University
Ottawa, Ontario, Canada

Figure 1: Re-rendering the content of one using style of another image.

ABSTRACT

The objective endeavour of this study is to mine available repositories of Image Stylization on GitHub and stack overflow posts, which influence the business of researchers since it is challenging to gauge the quality of work. In this study, we provide quality analysis of work explored and made available online as these repositories have never been examined as a community to identify unique properties, development patterns, and trends. We conducted an empirical study of Image stylization repositories hosted on GitHub (98) and stack overflow questions and answers (294) to produced evidence regarding popularity, methods, datasets, popular languages, ownership, safety, and trend analysis of available work that would be interesting for readers in the field when they choose the best methods for their work. We have captured the key insights of this community's history, such as its primary language (Python), most repositories are user-owned, created recently in 2020, deep learning and subjective evaluation methods are used widely, and no license information is provided for most repositories. Our findings show that the Image stylization community has unique characteristics that should be accounted for in future research.

CCS CONCEPTS

• **Computer science** → **Computer software**; • **Digital image processing**; • **Image Stylization**;

KEYWORDS

Mining software repositories, Neural networks, Image stylization, GitHub, Stack overflow

ACM Reference Format:

Ikram Hussain and Projna Saha. 2020. Large Scale Empirical Study of Image Stylization Repositories hosted on GitHub and Stack Overflow. In *Proceedings of December 15, 2020 (MSR Fall 2020).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the last few years, Image stylization has emerged as a hot topic since it provides a computer-based solution to stylize the content of one using the style of another image. Manual re-drawing of an image in an artistic style takes a long time for a professional artist. The seminal work of Leon Gaty's [4], which transfers the artistic style of a painting to real-world photographs, exemplified in Fig. 1, has opened up an interesting application area for deep networks and has triggered lots of follow-up works. This work was the first which use neural networks for stylization purposes, using the transfer learning technique. This approach uses the statistics of high-level feature representations of the images from the hidden layers of an image classification network to separate and reassemble content and style and received much attention inside and outside the computer vision community and triggered a whole line of research on deep learning based style transfer. Oftentimes, research activities are conducted by decentralized teams of programmers, students, and supervisors who cooperate globally using web-based source code repositories. In recent years collaborative works have proved

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MSR Fall 2020,
,
© 2020 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

their ability to produce high-quality software. A leading example of such a network is the GitHub website, an online social network that supports the development of software by virtual teams. Every GitHub user can create their repository and work on it with other registered users. They may also join projects created by another party and make their contributions there. Every GitHub user makes his/her own decision on how to manage their time and professional skills - that is the reason why the process of team formation on GitHub is decentralized and might consider as self-organizing work. We used this platform, where the tremendous work being done by the researcher and made available online for re-using and modification purposes, to gather all the Image stylization related contents and conduct our empirical study. Furthermore, Stack Overflow is a widely visited community Q/A site very popular with programmers and researchers. Research shows that over 92% of questions are answered in a median time of 11 minutes [8]. Programmers, irrespective of expertise levels, from beginners to expert level, do constructive discussions on this site. The questions are asked by the users and experts from the community come with possible answers among which one is marked as accepted by the user who asked the question originally. The registered users have ratings also based on the various services to the Stack Overflow community according to rules and regulations. We aimed to gather all posts related to image stylization especially and conduct our studies.

The remainder of this paper is as follows. Section 2 introduces the methodology, followed by finding the key insights, including data collection and filtering. Section 3 presents results and discussions for the findings. Section 4 present the threats to validity and implication of the work. Finally, Section 5 concludes the paper with future research directions.

2 METHODOLOGY

Fig. 2 shows the general framework of our methodology, starting with data collection using Image stylization keyword, followed by data filtering, cleaning and processing for capturing the key insights and challenges.

2.1 Data collection

We adopted the quantitative analysis of software repositories on GitHub, and gathered the relevant data, including the name, last update date, number of stars, issues, pull requests, commits, watch, forks, tags, contributors, languages used, license information, ownership and packages of all the available repositories as the sample size is not large. Having the research interest and knowledge about the field, we further mine if it is related to a publication, for instance, if someone's work has implemented or own work is available online. Dataset they have used, methods followed, such as deep learning or traditional stylization methods have been used, quality evaluation metrics used like if they used subjective quality assessment that is most favourable for evaluation purpose that includes human expertise, or objective assessment namely PSNR, SSIM, and VIF metrics. By the date 27 September 2020, 98 repositories are hosted on GitHub, summarized in Table 1.

Also, for stack overflow, we collected the data, including stack (question and answers) name, time and date of creation, time and date answer, tags, number and votes for answers, and the number of views. By the data 10 October 2020, 294 stack overflow questions and answers have been found.

Table 1: Analytics of repositories hosted on GitHub

Repositories	98
Code	31k
Commits	2k
Issues	428
Topics	1

2.2 Data filtering

We filtered out data using the content and size (i.e., non-empty). We eliminated non-relevant data using the tags of repositories and stack overflow questions and answers that resulted in two empty and two non-relevant (content-wise) repositories on GitHub and 270 questions and answers (Q/A's). These non-relevant repositories are filtered out, which were related to web development. The stack overflow search yielded most of the Q/A's related to web development. We identified the non-relevant data using the tags used in stack overflow platforms such as HTML, CSS, PHP for web development and TensorFlow, Pytorch, OpenCV for stacks related to image stylization. We did not filter out the data having zero stars as we have only 38 repositories having stars greater than one, but we excluded empty repositories.

2.3 GitHub Data

2.3.1 Trend analysis. We aim to find the origin of image stylization by looking at the oldest repository created. Boom period; in which year the topic attracted readers and researchers and most the repositories are created on GitHub. Also, the growth rate by the number of repositories per year and how the trend followed till the newest available.

2.3.2 Popularity. The popularity of a repository is judged by the Star power tool, the number of stars of each repository. In open-source software-based research, the number of stars is used as a filtering metric [6] [2].

2.3.3 Publication. We dig more into each repository and find out if a repository is created with the intention of implementation of someone's work, or own publication's implementation is made available, or own work related to image stylization is made available online. Moreover, we read the readme file of each repository to see if the owner gave information about the related publication. Further, we investigate if the paper is a conference or journal paper. For this, we study in-depth each repository and manually check the source of publication.

2.3.4 Methods used. With the boom of AI, machine learning and deep learning techniques, image stylization got boosted and especially after the breakthrough of Leon Gatys publication, which was the first to introduce the possibility of using neural networks for style transfer. In the recent past, traditional techniques were used for styles transferring, such as segmentation and geometrical shape-preservation, but nowadays, transfer-learning techniques

Unpublished working draft.
Not for distribution.

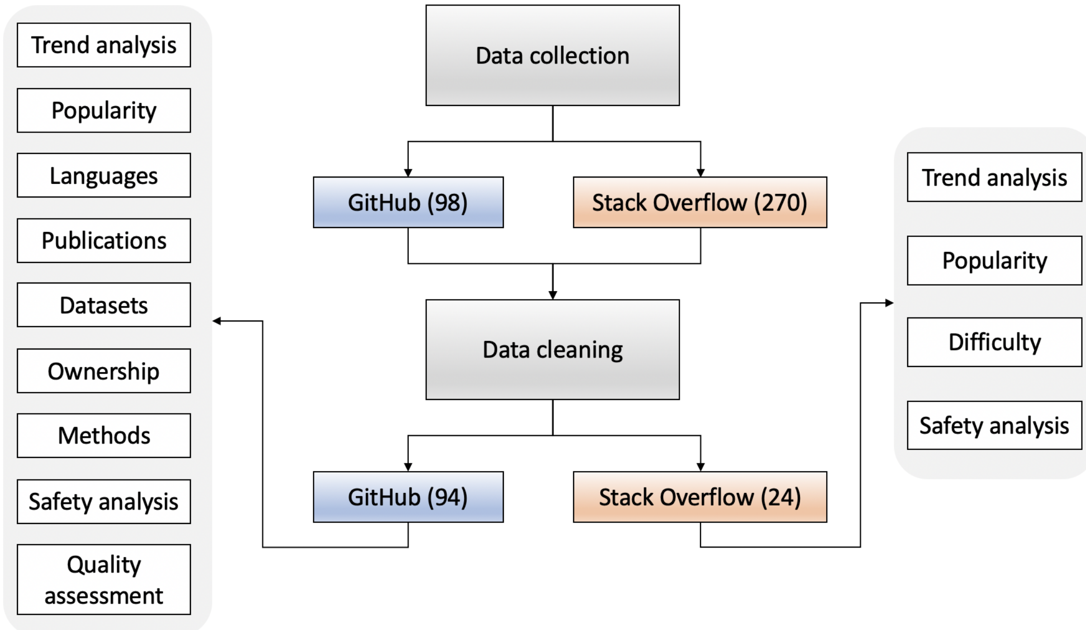

Figure 2: Framework of Methodology adopted

Unpublished working draft. Not for distribution.

that make use of trained models on millions of images, such as ImageNet and CIFAR10, are widely used for style transfer. It should be noted that the use of the neural network is still not explored enough and consider as a black box, which means that researchers are still not sure of why VGG16 is producing more appealing results than ImageNet. We use machine learning techniques to classify the methods used in repositories, such as traditional or deep learning-based. All the classifiers are tested using the Five-fold cross-validation. Dataset is first divided into train and test using 0.8 and .20.
Max-likelihood: First, the mean and covariance are calculated for 'n' features, mean matric of the shape nx1 and covariance nxn diagonal matrix, they are used to calculate the co-efficient of discriminant function and discriminant functions if the value is greater than zero assigns it to class one and vice versa.
Bayesian: Covariance is used the same one as of max-likelihood, but the mean is updated for Bayesian. The mean of Bayesian is calculated using equation (1).

$$\sigma(\sigma + 1/n\Sigma)^{-1} mean + 1/n\Sigma(\sigma + 1/n\Sigma)^{-1} Mo \qquad (1)$$

Where the absolute mean (Mo) is assumed as zero and absolute covariance, a matric of ones, they are used to calculate the co-efficient of discriminant function and discriminant functions, if the value is greater than zero assigns it to class one and vice versa. Here 'n' represents the number of samples in the dataset.
Fisher Discriminant: The weight matric 'v' is calculated using inverse (sum (covariance of 1st and 2nd class)) multiplies with mean of class1 minus mean of class2. The v matric has a shape of nx1 for n features of the data. We are finding a projection to a line to make both classes well separated. The weight matric can be multiplied with class1 and class2 to get the corresponding y1 and y2 matrices. After that, we can check if y1 values are more than zero then assign them to class 1 else class 2.
Ho-Kashyap: 'a' and 'b' metrics are arbitrary chosen of ones at the start the weight matric 'a' is updated each time until one of the three conditions become true. Firstly, if the no of iterations 'k' becomes equal to k-max, where k-max is chosen by the user. Secondly, if the b value is no more updating (It is noted, in Ho-Kashyap that our b value is not more updating after 5870th iteration, and so on for other K-Folds.), and lastly if Ya becomes greater than zero. Where Y matric is measured by combining classes one and two by adding a column of ones at the start. The learning rate (meow) is chosen the value of 0.9, used the calculation of b value using the equation B = b + meow [e + |e|]. Where e = Y*a − b. The weight matric a is calculated using the equation a = (Yt Y)-1 B. The weight matrix is used for the prediction of testing points.

2.3.5 *Quality evaluation methods.*

Evaluation methods used in image quality is a critical part. For instance, the peak signal to noise ratio (PSNR) is criticized by experts and is no more favourably used for image quality evaluation. Subjective assessment is considered as a 'king' of quality evaluation as it excludes the case of algorithmic errors, but subjective methods are not practical for a large amount of data. On the other hand, objective quality assessment is suitable for millions of images to report the average or BD-rate in case of image compression because of their convenience. Hence, experts in the field are proposing new metrics.

2.3.6 *Popular languages.*

A plethora of programming languages is used nowadays for various fields. We take into account all languages used in GitHub repositories to know which languages are frequently used for the problem at hand. This helps to understand the efficiency of a language for a specific task and especially for the young researchers, who are starting their careers.

2.3.7 Tags analysis. Tags are used by search engines to pop-up the relevant search results. We used the tags provided by the owners to identify if the content is related to image stylization. For example, HTML, CSS, and PHP are used for web development projects. We filtered out the data, which is not related to image stylization. Also, tags show packages or libraries used such as CMake, OpenCV, TensorFlow.

2.3.8 Safety analysis. We analyze the authenticity of the work by the license information of each repository such as MIT and Apache 2.0. Since available work may not be accurate or up to the mark, as it is not reviewed or peer-reviewed like the publication process, hence one might hesitate about re-using the work.

2.3.9 Ownership of repositories. To know about the origin of the work and which community or organization is contributing and investing more in a field, we studied the ownership. This information leads to knowing about the godfathers of the work and active members in the area.

2.4 Stack Overflow Data

2.4.1 Trend analysis. We measured the trend of stylization-related posts, including the oldest question asked, growth rate, and boom period. This gives us information about the origin of the work as well as the challenges faced over time in the community as a whole.

2.4.2 Popularity. Oftentimes, the popularity of a post is measured by the number of stars, but studies have shown a higher correspondence with the number of views and number of stars. Thus, we took into account the number of views for a particular question for popularity level.

2.4.3 Difficulty level. We measured the difficulty level of stacks if the question is not answered. Since all questions in our sample data are answered, so we checked how long it took to answers each question. Thus, we counted the number of days after each posted question until the answer.

2.4.4 Safety analysis. We have used the number of votes for each answer for the safety analysis of stacks. An accepted answer from one who asked the question also voted by other people shows the reliability of the answer and encourages others who have the same question.

3 RESULTS AND DISCUSSION

3.1 GitHub repositories

3.1.1 Trend analysis. Fig. 3 shows the trend analysis of image stylization repositories hosted on GitHub, it can be seen that the oldest repository was created in 2012, the trend is increasing with time, and 2020 has been noticed as a boom period that shows that this topic is under discussion in the research community and many repositories have been created.

3.1.2 Popularity. We measured the popularity of the repositories based on the star powers tool. Fig. 4 shows the top ten repositories

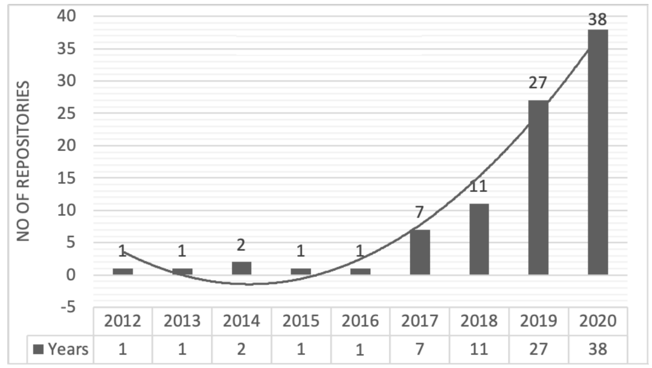

Figure 3: Trend analysis of image stylization repositories hosted on GitHub

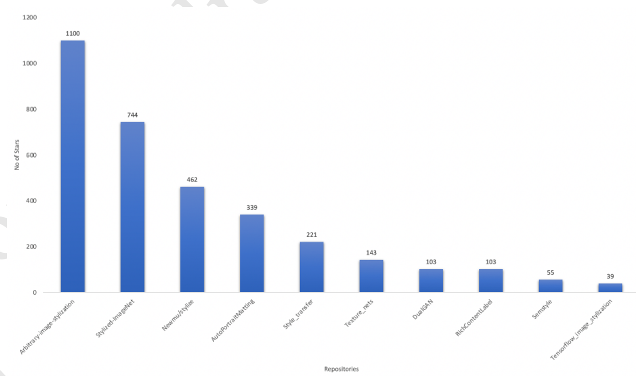

Figure 4: Top ten popular repositories hosted on GitHub

on GitHub with the highest popular topic, having more than 1000 stars, of arbitrary image stylization.

3.1.3 Publications. We research a unique way to find if a repository is related to a publication or not. It has been found that out of 96, 58 are related to publication, either their paper or someone's work has been implemented and made available as shown in Fig. 5. The rest 38 repositories contain their work but are not related to a publication. Furthermore, it is found that all the papers are conferences, and no journal paper has been found. This result helps the researcher community to make use of online repositories and reproducibility. The new researcher at the start of their work does not have to implement someone's work from scratch but can reproduce the work and use it for further research studies.

3.1.4 Methods used. We study the methods, traditional or deep learning in each study. This part of the study is the most debatable because the community as a whole is shifting from using traditional methods to state-of-the-art deep learning methods. We created the dataset containing 94 repositories and observed using tags such as VGG16, VGG19, ImageNet, transfer leaning as well as manual inspection and accordingly label the repository's methods used. Furthermore, we did one hot coding of string characters to convert into binary and used various machine learning classifiers to classify which methods are used in each case. Table 2 shows that the highest

Unpublished working draft. Not for distribution.

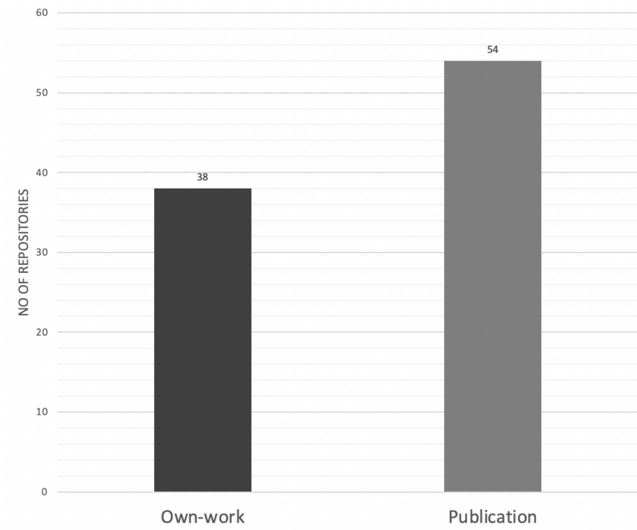

Figure 5: Repositories hosted on GitHub related to publications

accuracy is achieved, using KNN and stochastic gradient descent with the chi-square feature selection method. We used the Python scikit-learn library for these classifiers.

We also implemented four traditional classifiers, and preprocess the data using diagonalization. The highest accuracy achieved is for the Bayesian classifier then for max likelihood shown in Table 3. The reason is these two are quadratic discriminant classifiers/Analysis (QDA) and can classify the data more accurately. QDA treats both the classes as having its covariance matrix. Consequently, quadratic is more appropriate and able to find the non-linear decision boundaries by capturing the differing covariances of each class. Fisher and Ho-Kashyap are linear classifiers. The dataset is not easily separable because the means of both classes are close enough. Linear discriminant analysis (LDA) assumes the classes are gaussian and share the same covariance matric. Besides, the effect of feature selection has been analyzed and found it affects the accuracy of fisher discriminant, and thus the highest accuracy is reported. Also, Ho-Kashyap is dividing the classes into 50/50 but since the number of samples in class one is greater than the no of samples in class two, so we are getting the accuracy of 66%.

Effect of diagonalization: Diagonalization is also called whitening transformation or sphering transformation, which is a linear transformation that transforms a vector of random variables with a known covariance matrix into a set of new variables whose covariance is the identity matrix, meaning that they are uncorrelated, and each has variance 1. The transformation is known as "whitening" because it changes the input vector into a white noise vector. Diagonalization does not affect the accuracy of the system, but it speeds up the classification process. That is useful when we have significant data, and we want to classify then diagonalization is very helpful in the fast calculation.

Class imbalanced problem: One problem with the data is a class imbalance, which means the no of samples in one class is different from the other. This problem has also been addressed in the literature, and various solutions have been proposed so far. One solution to address the class imbalance problem is to generate artificial data that mimic the behaviour of real data. For example, SMOTE a python library provide a solution to upsample the minority class, also deep learning techniques can be used, such as GAN, style GAN, and DCGAN, through which artificial data is generated and can be used to address the class imbalance problem, for instance, this person does not exist, generate an artificial image each time the browser is refreshed. Also, evaluation using the other quality matrices is a useful solution.

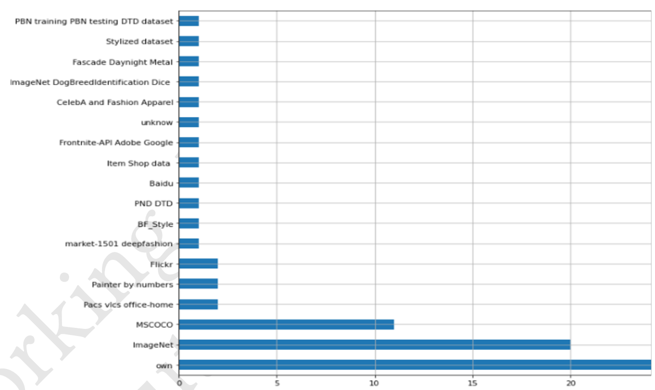

Figure 6: Datsets used in GitHub repositories

3.1.5 Dataset used. With the growing popularity, researchers are interested in which dataset has been widely used. We further study the datasets used in each work. Fig. 6 shows that most repositories used their datasets means they propose a new dataset also with their study. ImageNet, trained on more than 14 million images by an AI researcher Fei-Fei Li who began working on the idea of ImageNet in 2006 and poster publish it at the 2009 CVPR conference in Florida [3], has been widely used for transfer learning techniques to re-render the content of one image with the style of another. The Microsoft COCO dataset [7] is the second widely used dataset in the image stylization community, and so on.

3.1.6 Quality evaluation methods. Image quality evaluation is a hot topic among researchers. Fig. 7 shows that 92.98 studies have used subjective assessment for their work, and only 2.2% used objective quality metrics. This result shows the concern of researchers and experts on objective quality metrics.

3.1.7 Popular languages. We have found that Python is the champion language used in most repositories. Fig. 8 shows the top ten popular languages used in most projects. The majority of projects are in Python, followed by Jupyter Notebooks, C++, C, and JavaScript. We also observed trends over time using the repository creation date, and language labels are applied to a repository. Python has been found the most common language for almost all the repositories for the last eight years. In 2012, the first repository was labelled as C++ and C.

3.1.8 Tags analysis. Fig. 9 shows the most frequently used tags in GitHub repositories, and it can be seeing that image stylization and transfer learning tags are widely used, followed by Neural style,

Unpublished working draft. Not for distribution.

Table 2: Accuracy for methods used with and without feature selection

Classification Method	Accuracy		
	Without feature selection	Chi-square	PCA
SVM	0.7368	0.7368	0.7368
KNN	0.6842	**0.7894**	0.7368
Adaboost	0.5789	0.7368	0.6842
Logistic Regression	0.6842	0.7368	0.7368
Gaussian Naïve Bayes	0.4210	0.3682	0.3684
Random Forest	0.6842	0.7368	0.6315
MLP	0.6842	0.7368	0.6842
Stochastic Gradient descent	0.6842	**0.7894**	0.6315
Passive aggressive	0.7368	0.5789	0.6842
Ridge	0.7368	0.7368	0.7368

Table 3: Accuracy for methods used before and after diagonalization

Classification Method	Accuracy	
	Before diagonalization	After diagonalization
Max likelihood	0.69	0.69
Bayesian	**0.70**	**0.70**
Ho-Kashyap Rule	0.66	0.66
Fisher's Discriminant	0.60	0.60

ed working draft.
for distribution.

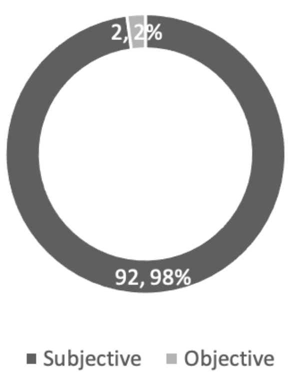

Figure 7: Quality evaluation methods used

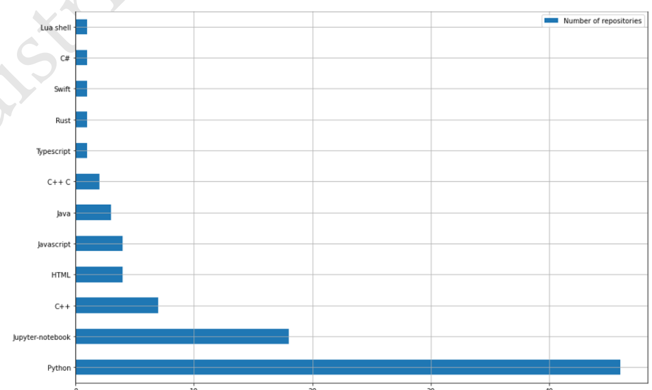

Figure 8: Top ten popular languages used in GitHub repositories

TensorFlow and Deep learning. This result shows frequently used tools and packages for Image style transfer projects.

3.1.9 Safety analysis. License information shows the right of users and owners to a repository. The MIT license is straightforward and open, it permits users to do anything with the project as far as they credit the authors and do not hold them liable for the project use, the Apache license is almost the same as MIT, but also grants patent rights to users [9]. Fig. 10 shows that most repositories do not have the license information, few (17%) have MIT license, fewer (10%) have apache-2.0 license, and only 1% have BSD 3 Clause license.

3.1.10 Ownership of repositories. Another factor is to know the collaboration environment if the owners are companies or individuals (single and multi). Table 4 presents that most repositories are singer users owned, then multi-users and surprisingly only one repository is owned by a company.

3.2 Stack overflow questions and answers

3.2.1 Trend analysis. Fig. 11 shows the trend analysis of image stylization post on stack overflow, and the oldest question was asked in 2012, the trend is increasing with time, and 2020 is the

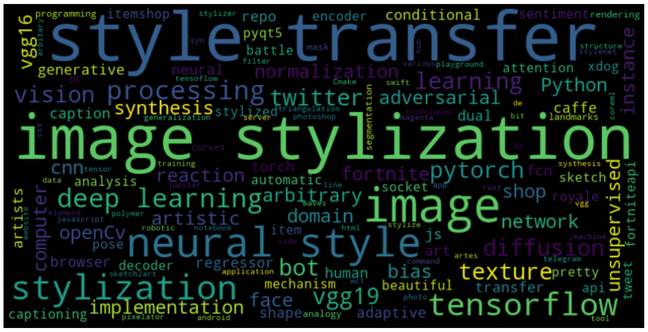

Figure 9: Frequently used tags in GitHub repositories

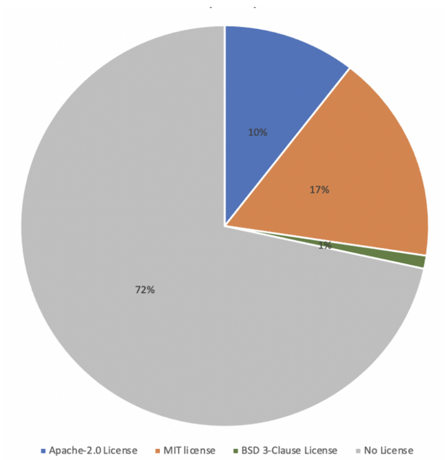

Figure 10: Safety analysis of GitHub repositories

Table 4: Analytics of Owner Types

Owner Type	Owner	Number of Repositories
Single	User	81
Organization	Company	1
Multi	Users	10

Unpublished working draft.
Not for distribution.

boom period that shows that this topic is under discussion in the research community and many questions have been asked.

3.2.2 Popularity analysis. Fig. 12 shows the top five popular posts on stack overflow, and it can be noticed that instance vs batch normalization has been discussed widely, followed by OpenCV rectangle with the dotted and dashed line and Image segmentation techniques.

3.2.3 Tags analysis. Fig. 13 shows the most frequently used tags in stack overflow posts, and it can be seen that Python and Image tags are widely used followed by TensorFlow, OpenCV, and neural. This result shows the popular languages, packages, and libraries used for this work.

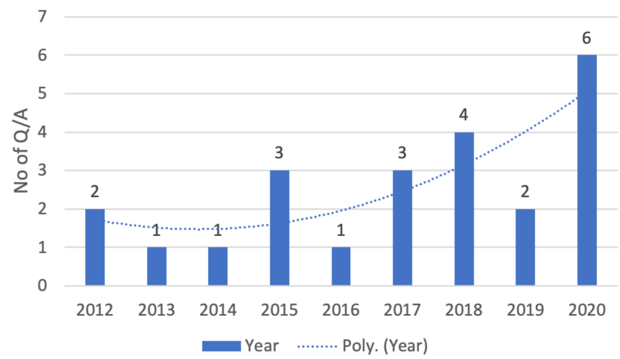

Figure 11: Trend analysis of stack overflow posts

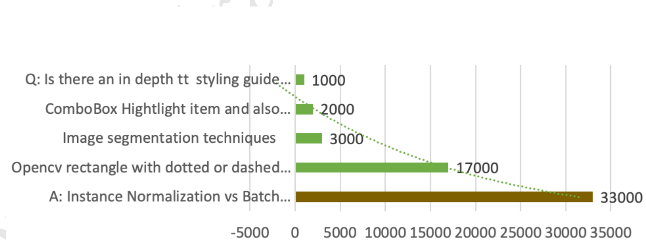

Figure 12: Top five popular stack overflow posts

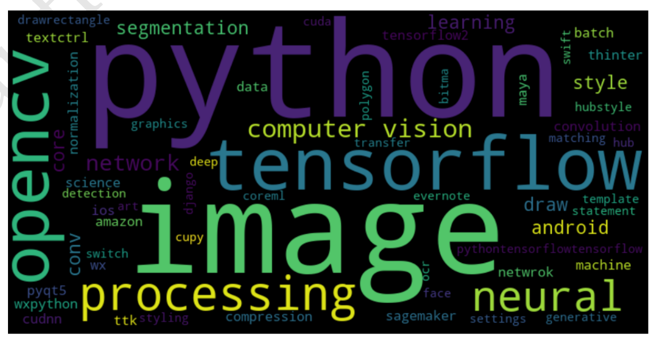

Figure 13: Tags analysis of frequently used tags in stack overflow posts

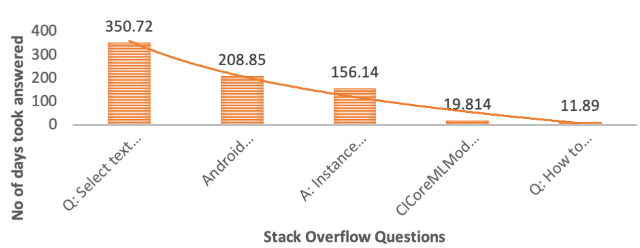

Figure 14: Top five difficult stack overflow questions

3.2.4 Difficulty analysis. We measured the difficulty level of questions on stack overflow, shown in Fig. 14 and interestingly the longest time to answer a question comprises almost a year, and the second-longest time taken is 208 days since posting. This result shows an exception to the studies [8] that claims an average time taken to a question is about 11 minutes.

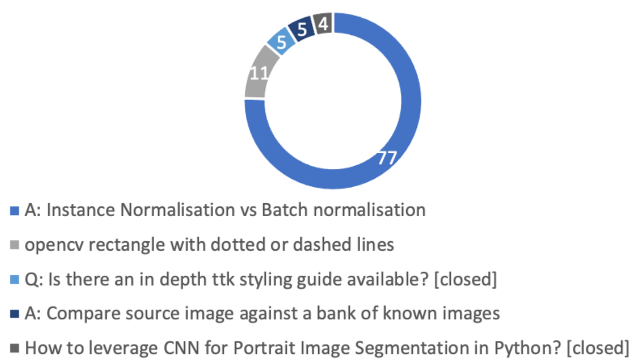

Figure 15: Number of votes for answers

3.2.5 Safety analysis. The owner of the post is the original user who posts the question, and experts from the community come with possible answers among which one is marked as accepted by the user who asked the question. Fig. 15 shows that the widely asked question has been voted the most (77 times), followed by the second most popular post. These results are correlated with the popularity of each question of stack overflow.

4 THREATS TO VALIDITY

Sample validity, a potential threat to the validity of a research study conducted on a small sample of subjects is that it could deliver little knowledge. We take into account all available repositories and stacks on GitHub and Stack overflow and do not further filter the data except using content and size (must be > 0) still the size of sample data is not significant. However, our sample consists of all available online repositories and stack overflow questions and answers. We have measured the difficulty of stack overflow questions if 1) the question is not answered and 2) no of days it took to answers. However, there could be other reasons contributing, such as fewer experts in the area.
Construct validity refers to the extent to which the operational measures in the work represents what we intended to measure. To identify the Image stylization projects, we relied on the topic labels that owners used to assign to repositories. That may lead to false positives and false negatives; some projects may not be assigned the correct labels, and as a result, incorrectly included or excluded from the sample data. To mitigate this risk, we double-checked manually if the tags/labels with the content of the repository. Furthermore, we used various machine learning classifiers to classify the methods used.
External validity is concerned with to what extent the findings of the work can be generalized. We focused the analysis on publicly available repositories on GitHub. The results might differ for private GitHub repositories, other code hosting sites, or projects in companies, for instance, Adobe research team Adobe research team. While we expect to see some differences, we also expect many similarities since some of the most popular Image stylization repositories are hosted on GitHub.
Our work is closely related to this [5] work, and we followed the benchmark set for an empirical study on GitHub and [1] for stack overflow. Besides, we mine the stack overflow platform intending to compare and contrast the challenges and trends in the field. We also added new methods such as finding if a work is related to a publication, datasets and methods widely used in the field, and the quality assessment technique adopted. To the best of the author's knowledge, no studies have explored the Image stylization repositories and stacks for the challenges and commonly used tools in the community.

5 IMPLICATIONS

Our work is helpful for those who are interested in image stylization, and they will know what work has been done already. The readers will find an implementation of many published papers here and can use them for modification and further studies. Python language and deep learning techniques are widely used for image stylization though little academic research is found on the Python language and the shift in trends from traditional methods to deep learning as it can be resource expensive. Subjective tests are used for evaluation purposes, so there is a gap for researchers to propose more trusted objective quality metrics that correlate well with the subjective assessment. For safety analysis, those who create new repositories are encouraged to have license information that will help the readers about the rights to re-use and authenticity. Demo videos can help readers and enhance reproducibility as users experience difficulties using available work. Our findings reveal that only one organization created an online GitHub repository which shows the active members in the field, and an opportunity for organizations to explore more in this area. Implementations on online platforms such as Google Colab is convenient for users, as it excludes the case to follow the requirements and platform dependability, which will save an enormous amount of time, and nevertheless, enable reproducibility.

6 CONCLUSION AND FUTURE WORK

In this paper, we gave a comprehensive summary of Image stylization repositories and questions and answers on GitHub (98) and stack overflow (294). As a result of the Image stylization boom over the year 2020, it has recently become the most debatable topic in the community. We focused on the analysis of growth, ownership, popularity, safety, publication, method, and evaluation. Among other findings (in Section 3), we found that Python language and deep learning techniques with subjective assessments are widely used by most drivers of the projects.
Future work can be extended to online surveys of experts in the field to get their opinions about problems and difficulties they are facing while re-using other people's work.

ACKNOWLEDGMENTS

This paper is part of a project for the course Mining Softwares Repositories offered in Fall 2020 at Carleton University, Ottawa. We are thankful to Professor Olga for offering the course and teaching

Unpublished working draft. Not for distribution.

the in-depth concepts through fruitful discussions and sharing her valuable experience with high professionality and dedication.

REFERENCES

[1] Ahmad Abdellatif, Diego Costa, Khaled Badran, Rabe Abdalkareem, and Emad Shihab. 2020. Challenges in chatbot development: A study of stack overflow posts. In *Proceedings of the 17th International Conference on Mining Software Repositories*. 174–185.

[2] Hudson Borges, Andre Hora, and Marco Tulio Valente. 2016. Understanding the factors that impact the popularity of GitHub repositories. In *2016 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 334–344.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[4] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2414–2423.

[5] Danielle Gonzalez, Thomas Zimmermann, and Nachiappan Nagappan. 2020. The State of the ML-universe: 10 Years of Artificial Intelligence & Machine Learning Software Development on GitHub. In *Proceedings of the 17th International Conference on Mining Software Repositories*. 431–442.

[6] Georgios Gousios and Diomidis Spinellis. 2017. Mining software engineering data from GitHub. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 501–502.

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[8] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. 2011. Design lessons from the fastest q&a site in the west. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2857–2866.

[9] LILY HAY NEWMAN. [n.d.]. *What Coders Should Know About Copyright Licensing*. https://www.fastcompany.com/3014553/what-coders-should-know-about-copyright-licensing

Unpublished working draft.
Not for distribution.