Neural Style Transfer for Kannada Fonts

Saahil B Jain
Department of Computer Science and Engineering, *PES University*
Bangalore, India
saahiljain98@gmail.com

Jeevana R Hegde
Department of Computer Science and Engineering, *PES University*
Bangalore, India
jeevanahegde@gmail.com

Srinivas K S
Department of Computer Science and Engineering, *PES University*
Bangalore, India
srinivasks@pes.edu

***Abstract*—Indian languages currently have a limited set of fonts available to choose from. Creation or generation of new fonts is at present, a manual process where designers draw letters in different styles. Further software tools to trace our these manually drawn styles. This can tend to be a tedious and time consuming process. This paper proposes a method to transfer style from existing fonts in languages such as English to letters of the Kannada script generate those fonts for the Kannada language. This would benefit a large audience, especially targeting the local audience for any business or enterprise.**

Keywords—Style Transfer, Neural Networks, Generative models, Fonts, Kannada.

I. Introduction

Currently font generation is a manual process where designers use tools to draw letters in different ways. No current work exist in automating the generation of fonts for Indian languages like Kannada. This brings the need for our project which aims to automate the generation of new fonts for the Kannada scripts while simultaneously ensuring generation of maatras.

Designers use software drawing tools to draw and create each letter of the alphabet set. Not only is this process hard because the designers have to ensure style consistency among every letter, but it is time consuming as well.

Recent research have attempted to generate fonts for the English language, however they require a large amount of pre-existing fonts to start generating new fonts. This amount of data doesn't exist for Indian languages like Kannada, which have only about 4-5 available fonts.

Hence a different approach is required to generate fonts for Indian languages so that fonts can be generated without the need for pre-existing fonts. This is where our solution comes into picture where we can generate new style consistent fonts without the need for pre-existing fonts.

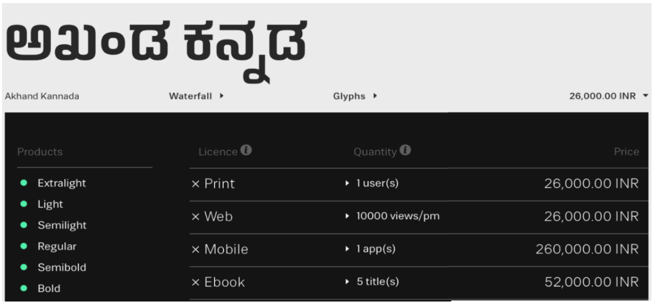
II. Literature Review

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organisational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

A. Background Study

In this chapter, we present the current knowledge of the area and review substantial findings that help shape, inform and reform our study. Before jumping into the research papers, we looked into why this problem statement is so necessary. That's when we looked into the current time in hours taken for each of it and further the costs from a particular website.

Taking cost into considerations, Indian Type Foundry (ITF) is a company which manually creates customised fonts for various languages. The charges range from 26,000INR - 2,60,000INR for a specific typeface.



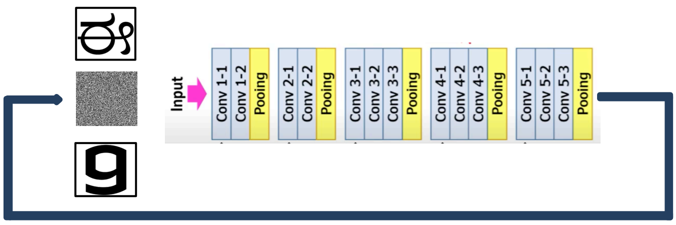
B. Architecture

In our attempt to achieve style transfer, we Initially tried out multiple methods including GANs and convolutional autoencoders.
However these methods had their own limitations. For example, GANs were dependent on a large amount of existing fonts already being available, which is not available for Indian languages such as Kannada. Convolutional Autoencoders were able to achieve style transfer to some degree, however the performance plateaued after a certain depth of the model and results were not satisfactory.

Hence we move to our approach of neural style transfer. Neural style transfer is an optimisation technique used to blend two images, a content image and a style image, such that the output image looks like the content image, but "painted" in the style of the style image.

Neural style transfer uses a pre-trained convolution neural network. To blend two images seamlessly to create visually appealing art, NST defines the following inputs:

- A content image (*c*) — the image we want to transfer a style to
- A style image (*s*) — the image we want to transfer the style from
- An input (generated) image (*g*) — the image that contains the final result (the only trainable variable)



To iteratively optimise the results we need to define a loss that can capture the structure of the content image as well as it can capture the style of the style image. Hence we split our loss function into two parts : the ***content loss*** function and the ***style loss*** function

The content cost function is making sure that the content present in the content image is captured in the generated image. It has been found that CNNs capture information about content in the higher levels, where the lower levels are more focused on individual pixel values [1]. Therefore we use the top-most CNN layer to define the content loss function.

$$L_{content} = \frac{1}{2} \sum_{i,j} (A_{ij}^l(g) - A_{ij}^l(c))^2$$

Since style information is measured as the amount of correlation present between features maps in a given layer. The style loss is defined as the difference of correlation present between the feature maps computed by the generated image and the style image. Mathematically, the style loss is defined as,

$$L_{style} = \sum_l w^l L_{style}^l \text{ where,}$$

$$L_{style}^l = \frac{1}{M^l} \sum_{ij} (G_{ij}^l(s) - G_{ij}^l g)^2 \text{ where,}$$

$$G_{ij}^l(I) = \sum_k A_{ik}^l(I) A_{jk}^l(I).$$

The total loss is a weighted sum of the content loss and the style loss.

$$L_{total} = \alpha\, L_{content} + \beta\, L_{style}$$

C. Previous Work

Recent research have attempted to transfer style for the English language, these use methods such as GANs to achieve the goal.

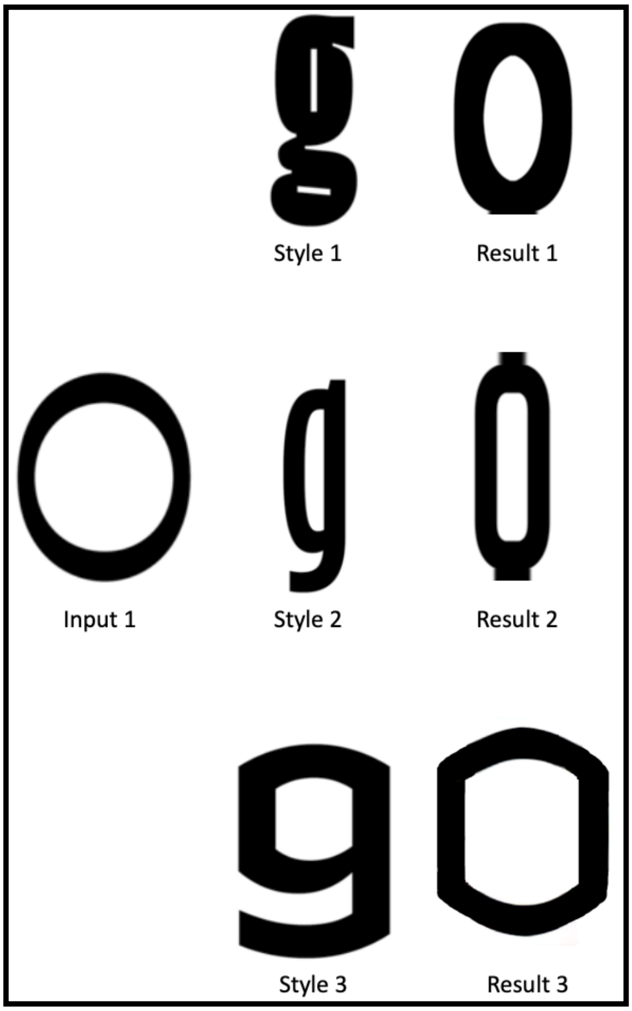
However these methods require a large amount of pre-existing fonts to start training the models. This amount of data doesn't exist for Indian languages like Kannada, which have only about 4-5 available fonts.

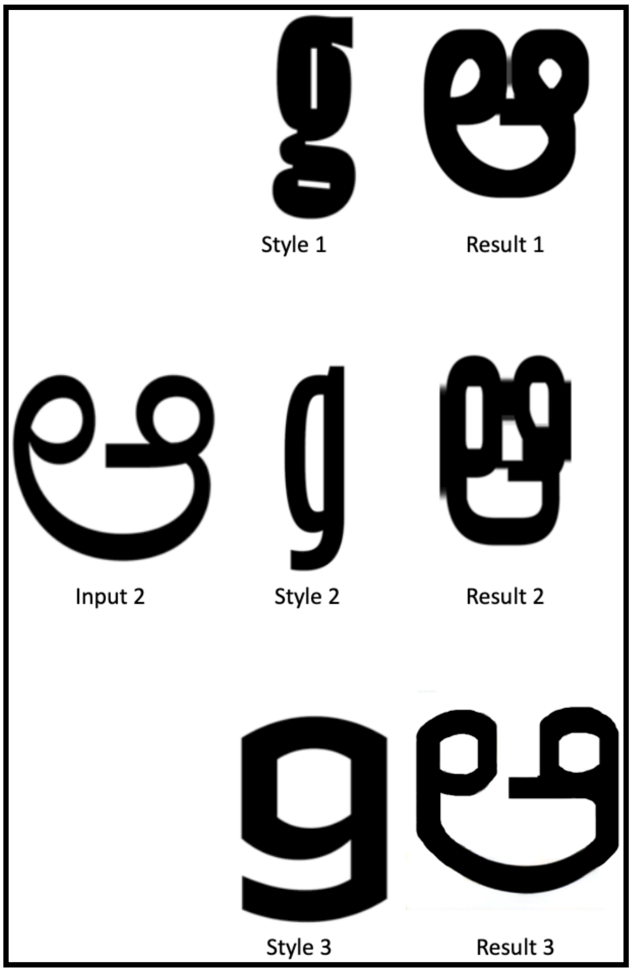
Hence we needed to use methods such as neural style transfer to achieve our goal without depending on vast amount of data.

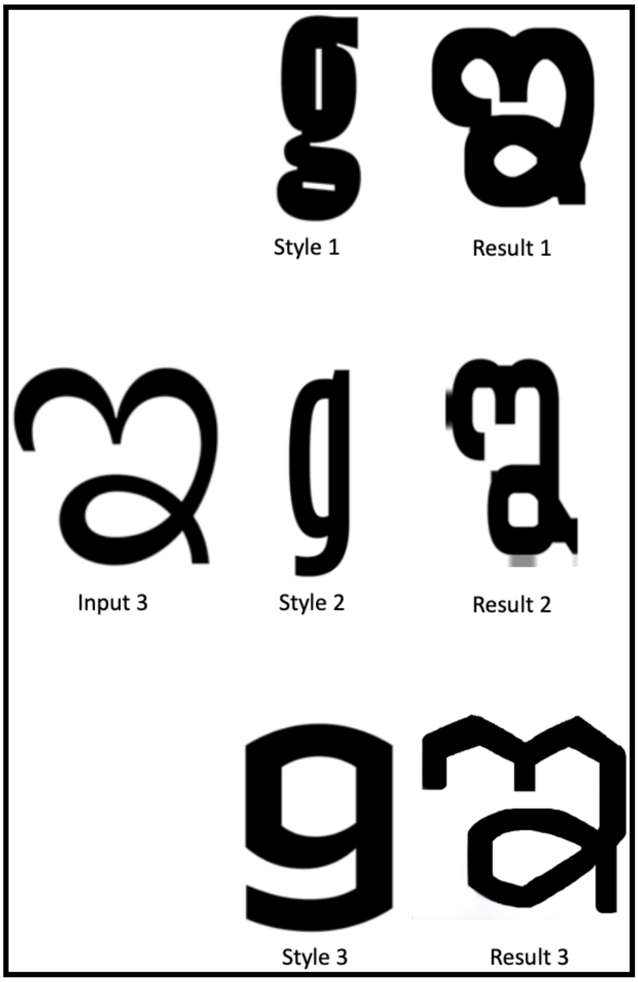
IV. Results and Discussion

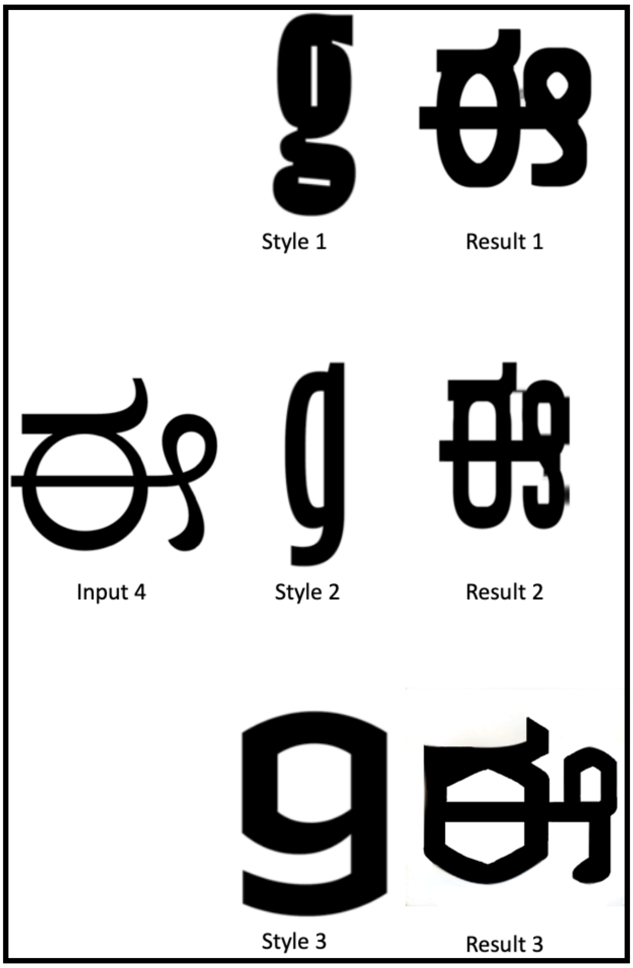
This approach is ideal for achieving style transfer when data availability is low. The generated results however do contain a little bit of noise. However this can be removed using image processing techniques.

The results of the approach are as follows:









V. Conclusion

We have successfully completed our goal to transfer the style of English fonts to Kannada.

Started off with deep diving into the literature survey once again to gain understanding on hw to work through the problem statement. This is where we did research on the different ways in which it can be done and came up with two approaches out of the many. These were:

1. Neural Style Transfer
2. Convolutional Autoencoders

For both these approaches, the dataset for English was required and hence we obtained the dataset for the same. We the cleaned and successfully labelled it to help with the mapping.

Convolutional Autoencoders were able to achieve style transfer to some degree, however the performance plateaued after a certain depth of the model and results weren't Satisfactory. Hence we used the Neural Style Transfer method .

We then built a model for both the approaches and trained and tested our data. The performance for autoencoders seemed to reach a limit and not get better beyond that point and hence we stuck to Neural Style transfer and carried it forward.

VI. References

1. Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, Trevor Darrell, "Multi-Content GAN for Few-Shot Font Style Transfer", Dec. 01, 2017. [Online] Available: https://arxiv.org/pdf/2006.06676.pdf [Accessed Jan. 16,2021]
2. Yonggyu Park, Junhyun Lee, Yookyung Koh, Inyeop Lee, Jinhyuk Lee, Jaewoo Kang, "Typeface Completion with Generative Adversarial Networks", Dec. 13, 2018. [Online], Available: https://arxiv.org/pdf/1811.03762.pdf [Accessed Jan. 05, 2020]
3. Yizhi Wang, Peking University, "Attribute2Font", May. 16, 2020. [Online], Available: https://arxiv.org/pdf/2005.07865v1.pdf [Accessed Jan. 22, 2021]
4. Leon A. Gatys, Alexander S. Ecker, Matthias Bathge, "A Neural Algorithm of Artistic Style", Sep. 02, 2015. [Online], Available: https://arxiv.org/pdf/1508.06576.pdf [Accessed Feb. 01, 2021]
5. Shuai Yang, Jiaying Liu, Zhouhui Lian, Zongming Guo, "Awesome Typography: Statistics-Based Text Effects Transfer", *Institute of Computer Science and Technology*, Dec. 06, 2016. [Online], Available: https://arxiv.org/pdf/1611.09026.pdf