PES UNIVERSITY

UE17CS490A – Capstone Project Phase – 1

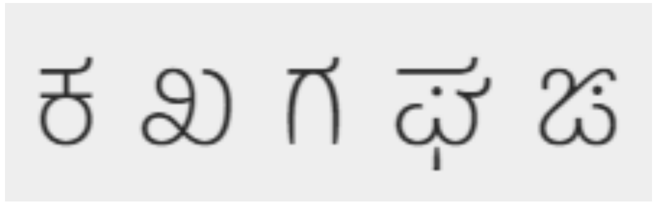
End Semester Assessment

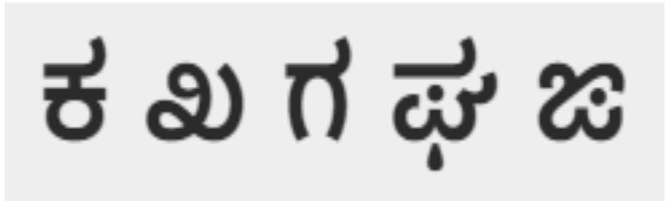
Project Title : Style Consistent Kannada Font Generation
Project ID : PW21KS04
Project Guide : Prof. K S Srinivas
Project Team : ~Saahil B Jain - PES1201700241
~Jeevana R Hegde - PES1201700633

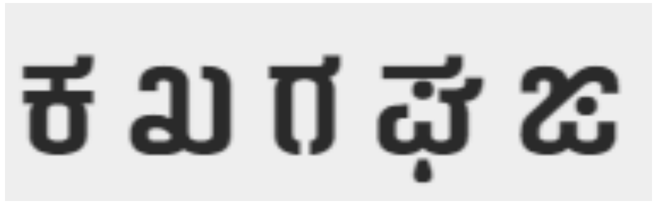
Agenda

Problem Statement

The project objective is to **automate** the generation of **new fonts which are style-consistent** for the **Kannada scripts** while simultaneously ensuring generation of maatras.







Abstract and Scope

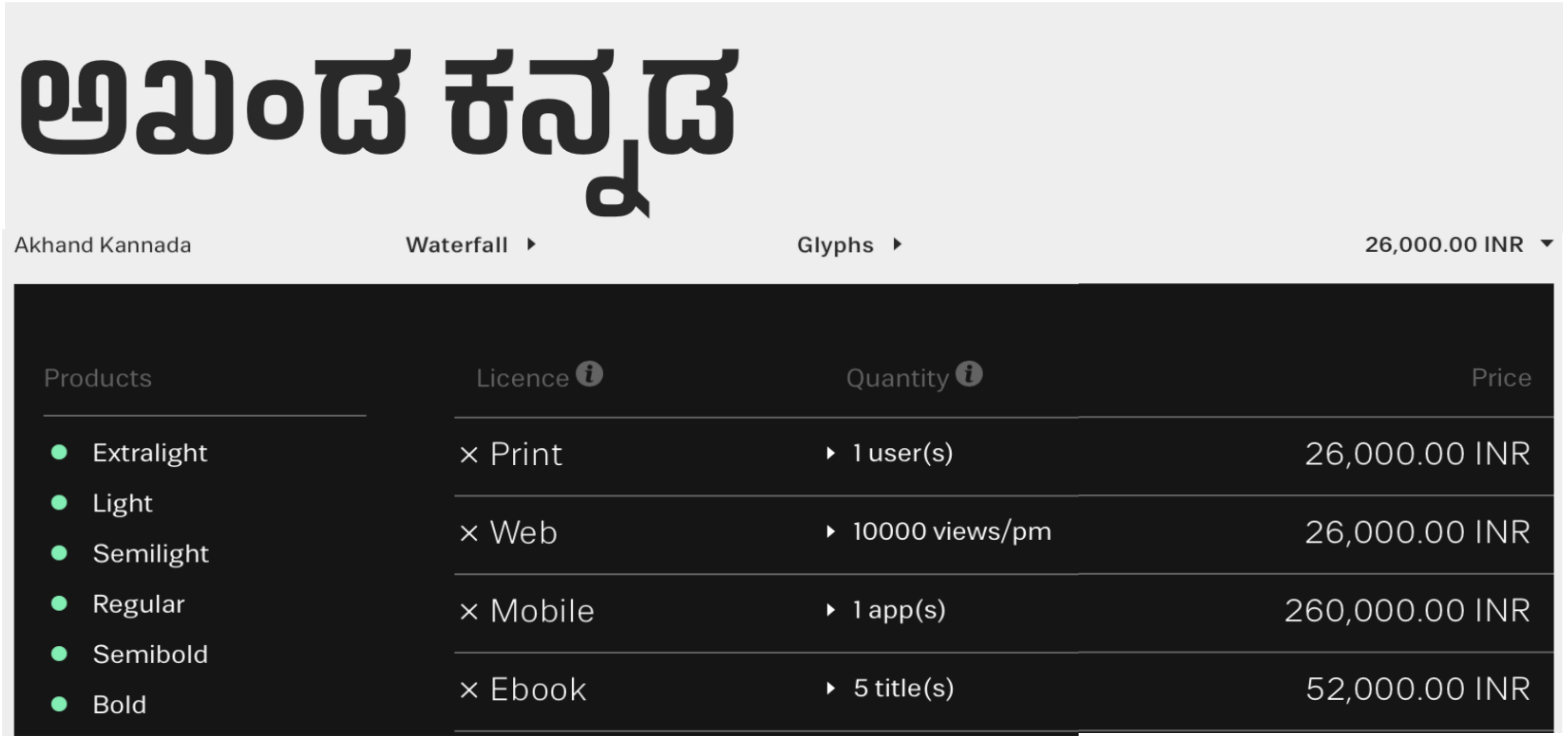
Abstract:

- **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.
- This can tend to be a **tedious, expensive** and **time-consuming** process.
- Our project goal is mainly to **automate** the generation of new fonts for the Kannada scripts while simultaneously ensuring generation of **ಮಾತ್ರಗಳು** (maatras) and **ಒತ್ತಕ್ಷರಗಳು** (otaksharas).
- This would benefit a large audience, especially targeting the local audience for any business or enterprise.

Abstract and Scope

Scope:

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



ಅಖಂಡ ಕನ್ನಡ

Akhand Kannada　Waterfall ▸　Glyphs ▸　26,000.00 INR ▾

Products

- ● Extralight
- ● Light
- ● Semilight
- ● Regular
- ● Semibold
- ● Bold

Licence	Quantity	Price
× Print	▸ 1 user(s)	26,000.00 INR
× Web	▸ 10000 views/pm	26,000.00 INR
× Mobile	▸ 1 app(s)	260,000.00 INR
× Ebook	▸ 5 title(s)	52,000.00 INR

Abstract and Scope

Scope:

- Currently font generation is done where the designers must **manually draw** and further **trace Glyphs** (characters) using software tools such as **Font Developer**.
- For each font, the time varies from a couple of hours to a couple of days.
- Our project aims to automate this process while simultaneously trying to incorporate style inspired from fonts available for other languages**(Style transfer)**.
- Hence makes the process **less time-consuming**, **more efficient(using style consistency)** and provides a **varied set of options to choose from at a cheaper price.**

PES UNIVERSITY

Literature Survey

The fonts we have seen are measured based on three factors:

•***Style consistency:*** The fonts that have the same style over all characters.

•***Legibility:*** The generated fonts are legible compared to the other methods.

•***Diversity:*** The generated fonts have a diversity which is different from the training images.

PES UNIVERSITY

Literature Survey

Research paper

[1] Eric Bernhardsson, “Deep-fonts”, Jan. 21,2016. Available: https://erikbern.com/2016/01/21/analyzing-50k-fonts-using-deep-neuralnetworks.html [Accessed Oct. 04, 2020]

Summary:

This paper planned to generate new fonts for English using simple neural networks. The neural network was trained on a dataset of 56,443 different fonts. This paper didn't generate very unique fonts, however it was one of the first papers to attempt this and was used as benchmark for other similar work.

Literature Survey

Strengths:

1. Enough data is present. Hence, slight modifications lead to the generation of significantly large number of fonts with high *legibility*.
2. *Style consistency* is reasonable.

Weaknesses:

1. There is an extremely high data dependency.
2. *Diversity* is quite low.

Literature Survey

Literature Survey

A B C D E F G H
I J K L M N O P
Q R S T U V W X
Y Z A B C D E F
G H I J K L M N
O P Q R S T U V
W X Y Z 0 1 2 3
4 5 6 7 8 9

Literature Survey

Research paper

[2] Hideaki Hayashia, Kohtaro Abea, Seiichi Uchidaa, GlyphGAN, *Department of Advanced Information Technology*, May 30, 2019.

Available: https://arxiv.org/pdf/1905.12502.pdf [Accessed Sep. 04, 2020]

Summary:

This is a paper that successfully developed a model for style-consistent font generation for English. It is based on deep convolution generative adversarial networks (GANs).

The data requirement is about 6561 different fonts.

This paper provides a significant benchmark to our project.

Literature Survey

Strengths:

1. Better *legibility* compared to deep-fonts.
2. *Style consistency* is the same.
3. More *diversity* in available in the fonts compared to deep-fonts.

Weaknesses:

1. High data dependency is present, which is unavailable for Indian languages like Kannada.
2. The concept of style transfer is not present.

Literature Survey

Similarities:

Method	Deep fonts	GlyphGAN
Goal	*New font generation*	*New font generation*
Style consistency	*0.47*	*0.47*

Differences:

Method	Deep fonts	GlyphGAN
Data requirement	*56443*	*6561*
Legibility	*72.51*	*83.90*
Diversity	*0.33*	*0.61*

Literature Survey

Article in Online Encyclopedia

[3] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, Michael Levin, "Growing, Neural Cellular Automata", Feb. 11, 2020. [Online],
Available: https://distill.pub/2020/growing-ca/ [Accessed Sep. 02, 2020]

Summary:
The goal is of emoji generation as well as regeneration using neural cellular automata.
This paper acts as the starter of how to get our project started. Hence plays a vital role in our project.

Literature Survey

Strengths:

1. Good emoji generation from just a single point.
2. *Regenerative* abilities is present i.e. can complete the emoji if part of it is missing.
3. Can generate fairly complex emojis.
4. Models are quite small.

Weaknesses:

1. No new generation as such. Just builds a model to create the exact input.
2. Every emoji requires a separate model.

PES UNIVERSITY

Literature Survey

Article in Online Encyclopedia

[4] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, Michael Levin, "Self-Classifying MNSIT digits", Feb. 11, 2020. [Online],
Available: https://distill.pub/2020/selforg/mnist/ [Accessed Oct. 07, 2020]

Summary:
The goal is to classify MNIST Digits using neural cellular automata. This is proposed by the same authors as the previous paper. However it focuses on classifying MNSIT Digits as opposed to generation and regeneration.

Literature Survey

Strengths:

1. Good digit classification which is the main goal.
2. It can identify digits within patterns as well.

Weaknesses:

1. Uses *colour* to distinguish. Hence, adding more classes reduces accuracy.

Suggestions from Review - 3

Suggestions and remarks given:

- Need to get a clear idea on our user class.
- To not limit it to only Kannada language.
- Highlight the existing methods and why our approach is necessary and better.

Design Details

Novelty

The novelty of our project lies not only in the fact that **automation of font generation is new for Indian languages like Kannada**, but also the use of technologies like Neural Cellular Automata for generation problems. **Neural Cellular Automata has never been used for generation problems**, Our project is the first use case of Neural Cellular Automata as a generative model.

Innovativeness

Neural Cellular Automata has often been used for recreational problems, but never for generation problems. Our project is the first use case of Neural Cellular Automata as a generative model. Our project is also **unique** because it is **able to generate new fonts without any data requirements.**

Design Details

Interoperability

Once the fonts are generated as images and converted to fonts usable by the system, **they can be downloaded on the system on which the application requires them.** Can be used an additional tool for any application once deployed. Once downloaded they can be used on literally any system as regular fonts are used.

Dependencies

1 set of glyphs/images of alphabets with all maatras for whichever language.

Design Details

Application compatibility

Provided **images in any image format** like jpg,png,etc., new fonts can be **enerated as images**. These **images** can easily be **converted as fonts** and used long with literally any application that requires these fonts.

Legacy to modernization

- Currently fonts are generated **manually** by designers. These 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.

Legacy to modernization (continued)

- 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**.

Design Details

Portability and reusability

Our method **can easily be used to generate new fonts for other languages ike Telugu or Hindi.**

It can also be used for many generative tasks as well.

For example: this method can also be **used for data augmentation ourposes for images which would be very helpful in case of low clarity mages.**

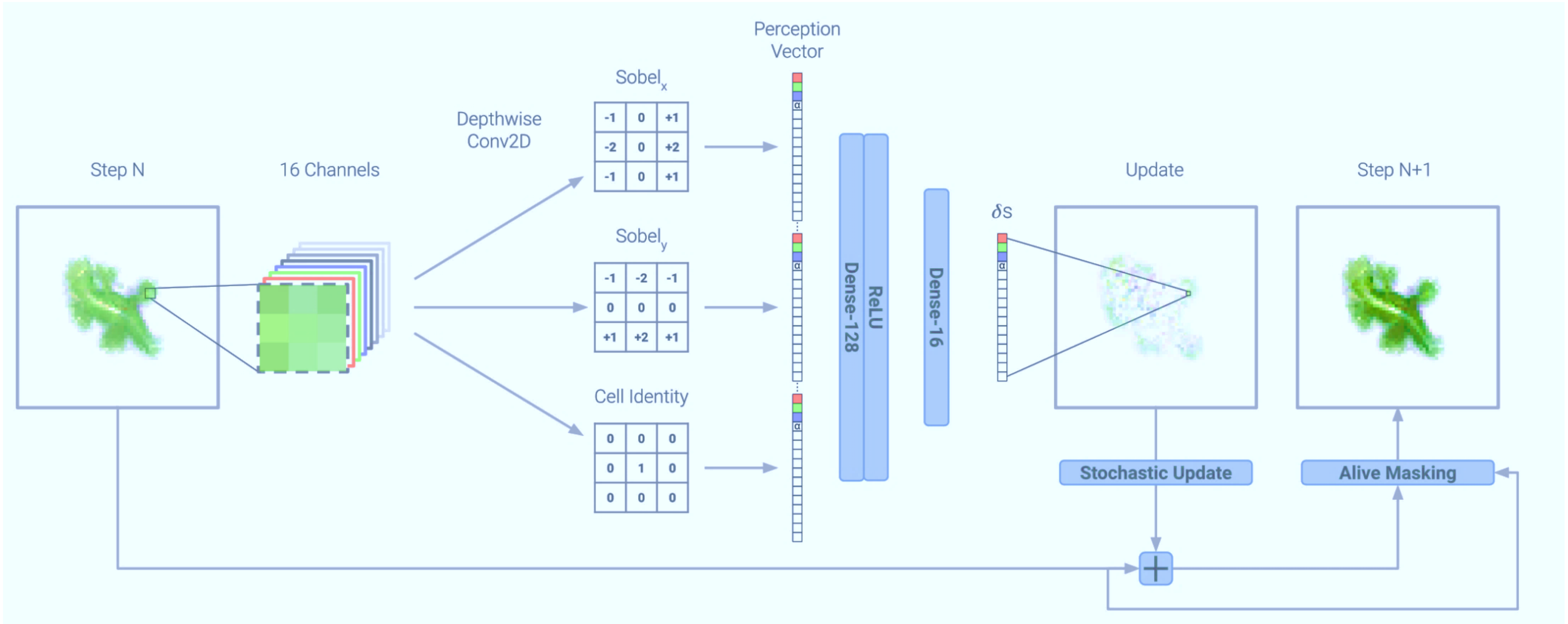
Constraints

To generate fonts we need to **create separate models for each letter** and separate model for each letter with different maatras as well. This makes generation of base models **challenging** as well as **time consuming**.

Processing power required is considerably low. But **increase in image size ncreases model size exponentially**.

Proposed Methodology / Approach

Proposed architecture:



A single update step of the model.

Proposed Methodology / Approach

Proposed approach:

- We plan to use ***Neural Cellular Automata*** to solve the problem statement.
- We will represent each cell state as a vector of ***16 real values***.
- Then we ***learn an update rule*** for each cell ***using its neighbours***.
- Applying this update rule ***multiple times*** will help generate the given font.
- ***Slight moderations*** in the feature extraction can result in ***unique fonts***.

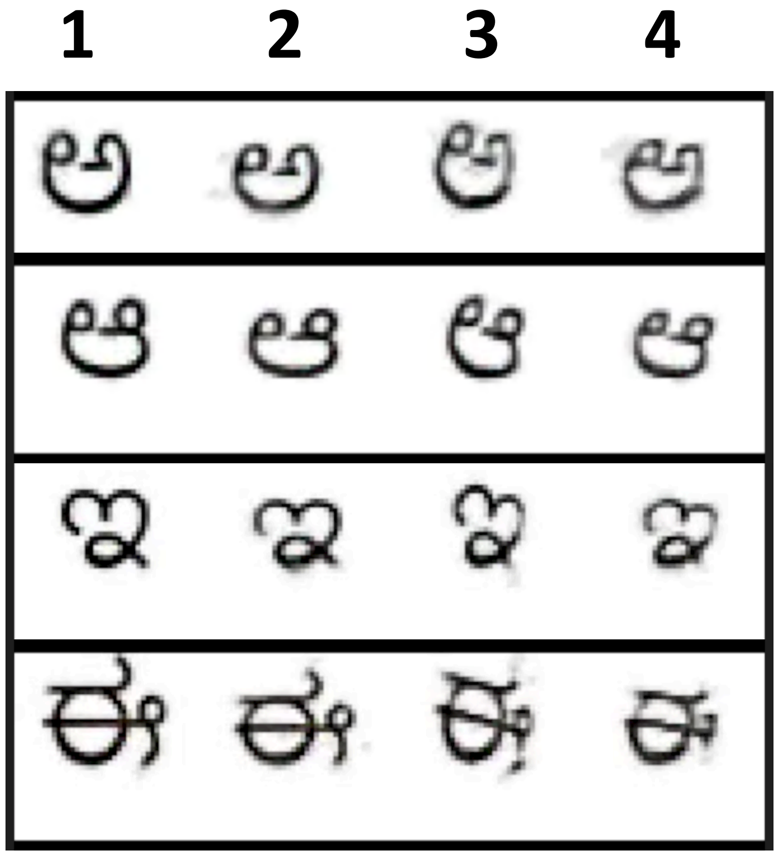
Demonstration:

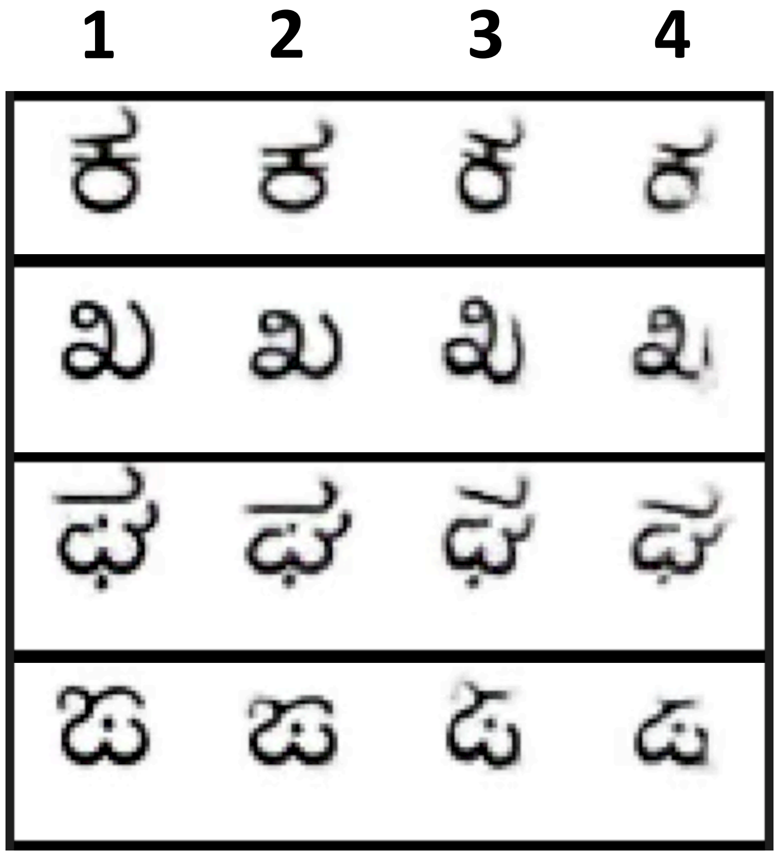
Variability

Proposed Methodology / Approach

Demonstration:

Style-consistency





Proposed Methodology / Approach

Benefits of this approach:

- Less time consuming as the process is automated.
- More efficient as style consistency is being taken care off.
- More options for a customer to choose from as numerous options can be created with the same set of input.

Drawbacks:

- Probably lesser customization for a user.

Technologies Used

The technologies / platforms/ languages used:

- Machine learning: to provide a variable update rule.
- Cellular automata: to generate without large data requirements.
- Python: to prototype faster.
- Google Colab/Kaggle Notebook: GPU power.

Project Progress

Project progress:

- We started off with doing a literature survey on the problem statement and background study on the work currently being done.
- We then deep-dived into the approach of Neural Cellular Automata to solve the problem and downloaded the data sets and cleaned them as required.
- Next, we worked on developing hidden channel and did feature extraction.
- We then built a basic model of Neural Cellular automata for a single letter and extended it to all letters. We also incorporated maatras for every letter.
- Currently generation is completed for the entire dataset and fine tuning is being done.
- We have also finalized the documentation for our research project.

So we would claim we have completed **100%** in terms of **phase 1**.
We are **65%** done **with respect to both phases** i.e. Including next semester.

References

Article in Online Encyclopedia

[1] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, Michael Levin, “Growing, Neural Cellular Automata”, Feb. 11, 2020. [Online], Available: https://distill.pub/2020/growing-ca/ [Accessed Sep. 02, 2020]

Journal Article Abstract

[2] Koen Janssens, “Network Cellular Automata—A Development for the Future?”, *Computational Materials Engineering*, 2007. [Online], Available: https://www.sciencedirect.com/topics/computerscience/cellularautomata[Accessed Sep. 03,2020]

Research paper

[3] Hideaki Hayashia, Kohtaro Abea, Seiichi Uchidaa, GlyphGAN, *Department of Advanced Information Technology*, May 30, 2019. Available: https://arxiv.org/pdf/1905.12502.pdf
[Accessed Sep. 04, 2020]

Research paper

[4] Eric Bernhardsson, “Deep-fonts”, Jan. 21,2016. Available: https://erikbern.com/2016/01/21/analyzing-50k-fonts-using-deep-neural networks.html
[Accessed Oct. 04, 2020]

Any other information

Summary:

Successfully built and trained a **Neural Cellular Automata** and updated it’s filters to generate **style-consistent** fonts for Kannada script.

1. Fonts generated are style-consistent.
2. Generates new fonts without requirements of multiple fonts(data).

The user class of this television channels, newspapers, magazines and books, movie names/posters, company name(specially targeted for local audience).

PES UNIVERSITY

Thank You