LOVELY PROFESSIONAL UNIVERSITY

School of Computer Science and Engineering Faculty of Technology And Sciences

NAME	REG.no	ROLL NO.
RAMPRAWESH KUMAR	11809515	09
MANIKRANTH REDDY	11809502	12
PAVAN KUMAR	11809498	11
PRASHANT KUMAR	NOT GIVEN	10

SECTION:- K18PB

GROUP: - GROUP1

Course title:-Artificial intelligence

Course code:-INT404

Faculty name:-NIKITA KAUSHIK

Date of submission:-08/04/2020

Github link :- https://github.com/Ramprawesh123/Artificial-intelligence

ABSTRACT :

Handwriting Detection is a technique or ability of a Computer to receive and interpret intelligible handwritten input from source such as paper documents, touch screen, photo graphs etc. Handwritten Text recognition is one of area pattern recognition. The purpose of pattern recognition is to categorizing or classification data or object of one of the classes or categories. Handwriting recognition is defined as the task of transforming a language represented in its spatial form of graphical marks into its symbolic representation. Each script has a set of icons, which are known as characters or letters, which have certain basic shapes. The goal of handwriting is to identify input characters or image correctly then analyzed to many automated process systems. This system will be applied to detect the writings of different format. The development of handwriting is more sophisticated, which is found various kinds of handwritten character such as digit, numeral, cursive script, symbols, and scripts including English and other languages. The automatic recognition of handwritten text can be extremely useful in many applications where it is necessary to process large volumes of handwritten data, such as recognition of addresses and postcodes on envelopes, interpretation of amounts on bank checks, document analysis, and verification of signatures. Therefore, computer is needed to be able to read document or data for ease of document processing.

Introduction:

The field of pattern recognition has become one of the broad areas . Optical character recognition (OCR) is one of the fields in pattern recognition; its purpose is to transform an image of handwritten, typewritten or printed text into an understandable representation that a computer can easily

recognize. Consequently, the OCR system is applied in several applications in various domains such as: bank check processing, postal code recognition, mail sorting, digital libraries, security system, etc. However, OCR system development is a non trivial task because the word have an infinite number of representations due to that each person writes with his own way, which is different from the others , and also in view of the fact that there are many font to print with many styles (bold, italic, underlined, etc), with different complex layouts. Depending on the type of writing that a system should recognize (manuscript, cursive or print), operations to be performed and the results can vary significantly. The OCR system is widely divided into two types:

1.On-line recognition and

2.Off-line recognition.

In the On-line type, the process of writing recognition is performed at the same time when the user is writing, while the off-line type is static recognition in which the writing recognition is carried out after completion of writing.

Generally, an OCR system is a mechanism that includes several stages for translating an image of printed or handwritten text into a form that the machine can manipulate .These stages are called: pre-processing , segmentation, feature extraction and classification. Probably the most well-known use case for OCR is converting printed paper documents into machine-readable text documents. Once a scanned paper document goes through OCR processing, the text of the document can be edited with word processors like Microsoft Word or Google Docs. Before the OCR technology was available, the only option to digitize printed paper documents was to manually re-type the text. Not only was this massively time consuming, it also came with typing errors.OCR is often used as a hidden or silent technology, powering many well-known systems and services in our daily life. It's used in data entry automation, indexing documents for search engines, automatic number plate recognition, as well as assisting blind and visually impaired people.

Literature REVIEW :

Character Recognition Algorithms

The algorithms used in character recognition can be divided into three categories: Image Pre-processing, Feature Extraction, and Classification. They are normally used in sequence – image pre-processing helps makes feature extraction a smoother process, while feature extraction is necessary for correct classification. Here's how they work:

Image pre-processing

Image pre-processing is crucial in the recognition pipeline for correct character prediction. These methods typically include noise removal, image segmentation, cropping, scaling, and more. The recognition system first accepts a scanned image as an input. The images can be in JPG or BMT format.

Digital capture and conversion of an image often introduces noise, which makes it hard to identify what is actually a part of the object of interest. Considering the problem of character recognition, we want to reduce as much noise as possible, while preserving the strokes of the characters, since they are important for correct classification.

Segmentation

In the segmentation stage, a sequence of characters is segmented into a sub-image of an individual character. Each character is resized into 30×20 pixels.

Classification and Recognition

This stage is the decision making stage of the recognition system. The classifier contains two hidden layers, using a log sigmoid activation function to train the algorithm.

Feature extraction

The features of input data are the measurable properties of observations, which is used to analyse or classify these instances of data. The task of feature extraction is to identify relevant features that discriminate the instances that are independent of each other.

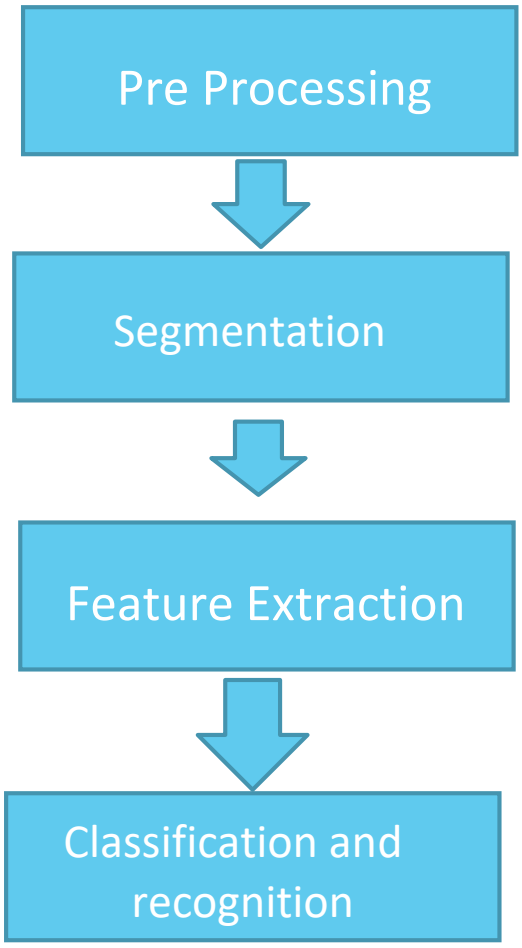
Proposed Methodology and Strategies

Recognition strategies heavily depends on the nature of the data to be recognized.

In the cursive case, the problem is made complex by the fact that the writing is fundamentally ambiguous as the letters in the word are generally linked together, poorly written and may even be missing.

On the contrary, hand printed word recognition is more recognition, the individual letters composing the word being usually much easier to isolate and to identify.

Methodology :



Important points to be considered for conversion

Text density

Structure of text

Fonts

Character type

Artifacts

Locations

How it works?

To convert the image into text we use OCR.

OCR stands for Optical Character Recognition.

It is a widespread technology to recognise text inside images, such as scanned documents and photos.

OCR technology is used to convert virtually any kind of images containing written text (typed, handwritten or printed) into machine-readable text data.

Probably the most well known use case for OCR is converting printed paper documents into machine-readable text documents.

Once a scanned paper document went through OCR processing, the text of the document can be edited with word processors like Microsoft Word or Google Docs.

Online recognition

On-line handwriting recognition involves the automatic conversion of text as it is written on a special digitizer or PDA, where a sensor picks up the pen-tip movements as well as pen-up/pen-down switching. That kind of data is known as digital ink and can be regarded as a dynamic representation of handwriting. The obtained signal is converted into letter codes which are usable within computer and

text-processing applications. The elements of an on-line handwriting recognition interface typically include:

- A pen or stylus for the user towrite with.
- A touch sensitive surface , which may be integrated with, or adjacent to, an output display.
- A software application which interprets the movements of the stylus across the writing surface, translating the resulting strokes into digit text.

Offline recognition

Off-line handwriting recognition involves the automatic conversion of text in an image into letter codes which are usable within computer and text-processing applications. The data obtained by this form is regarded as a static representation of handwriting. Off-line handwriting recognition is comparatively difficult, as different people have different handwriting styles. And, as of today, OCR engines are primarily focused on machine printed text and ICR for hand “printed” (written in capital letters) text. There is no OCR/ICR engine that supports handwriting recognition as of today.

Character recognition

Character recognition techniques can be classified according to two criteria:

The way preprocessing is performed on the data

The type of the decision algorithm

Preprocessing techniques include:

- The use of global transforms (correlation, fourier descriptors, etc.)

- Local comparison (local density, intersections with straight lines, variables masks, etc.)
- Geomaterial or topological characteristics (strokes, loops, openings, diacritical marks, skeleton, etc.)
- The types of the decision algorithm

Decision method include:

- Various statistical methods
- Neural networks, structural matching (on trees, chains, etc.)
- Stochastic processing (Markov chains, etc.).

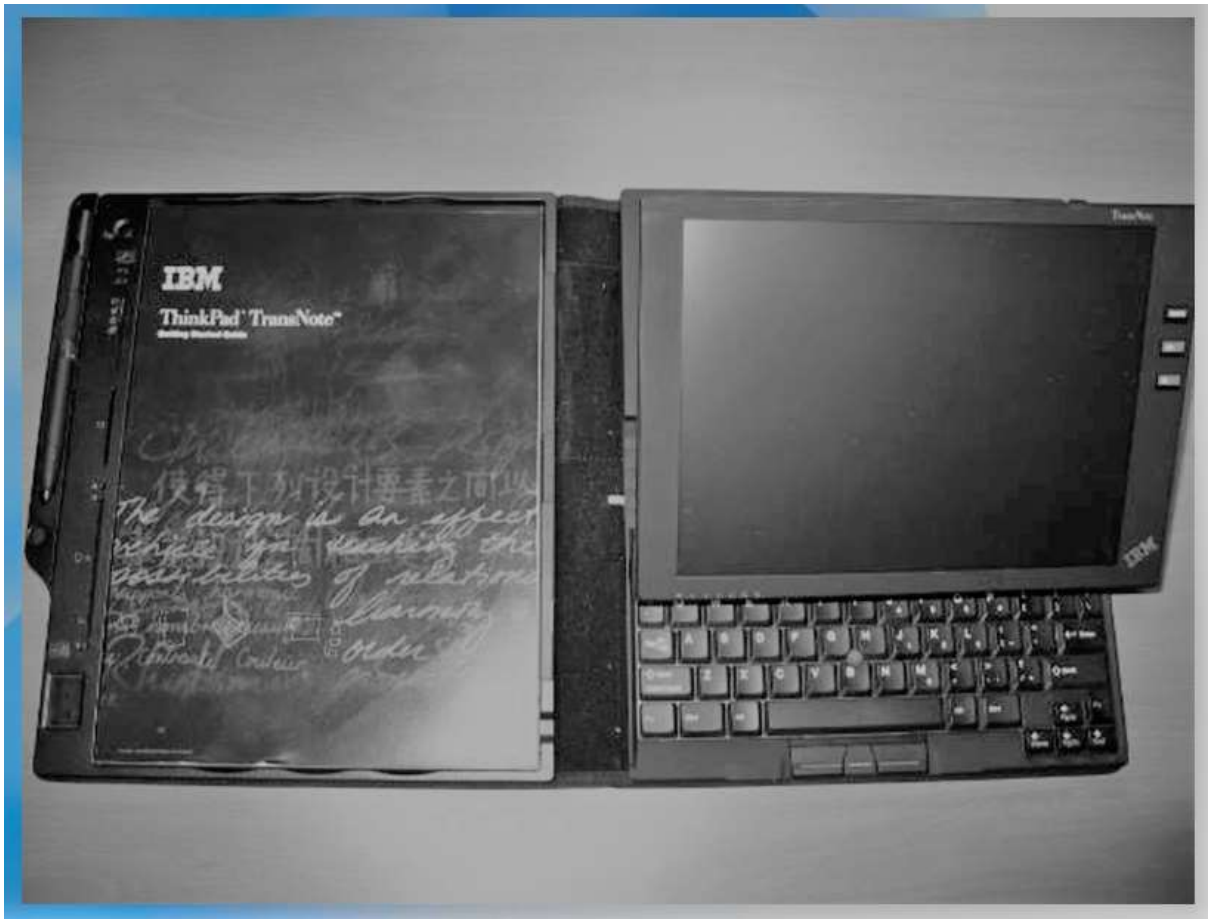
Word recognition:

Two main types of strategies have been applied to this problem

- The holistic approach – recognition is globally performed on the whole representation of words and there is no attempt to identify characters individually.
- The main advantage of holistic methods is that they avoid word segmentation
- The analytical approach:-deal with several levels of representation corresponding to increasing levels of abstraction (usually the feature level, the grapheme or pseudo-letter level and the word level). Words are not considered as a

whole, but as sequences of smaller size units which must be easily related to characters in order to make recognition independent from a specific vocabulary

An example : IBM transnote



Need of conversation:

Before OCR technology was available, the only option to digitise printed paper documents was to manually re-typing the text.

Not only was the massively time consuming, it also came with inaccuracy and typing errors.

OCR is often used as a “hidden” technology, powering many well known systems and services in our daily life.

Less known, but as important, use cases for OCR technology include data entry automation, indexing documents for search engines, automatic number plate recognition, as well as assisting blind and visually impaired persons.

Program for handwriting recognition:

Import matplotlib.pyplot as plt

```
import tensorflow as tf
```

import numpy as np

from sklearn.metrics import confusion matrix

```
from datetime import timedelta
import collections
import re, os, cv2
from glob import glob

character_mapping = {
0: 0, 1: 1, 2: 2, 3: 3, 4: 4, 5: 5, 6: 6, 7: 7, 8: 8, 9: 9, 10:
'A', 11: 'B', 12: 'C', 13: 'D', 14: 'E', 15: 'F', 16: 'G', 17:
'H', 18: 'I', 19: 'J', 20: 'K', 21: 'L', 22: 'M', 23: 'N', 24:
'O', 25: 'P', 26: 'Q', 27: 'R', 28: 'S', 29: 'T', 30: 'U', 31:
'V', 32: 'W', 33: 'X', 34: 'Y', 35: 'Z', 36: 'a', 37: 'b', 38:
'c', 39: 'd', 40: 'e', 41: 'f', 42: 'g', 43: 'h', 44: 'i', 45:
'j', 46: 'k', 47: 'l', 48: 'm', 49: 'n', 50: 'o', 51: 'p', 52:
'q', 53: 'r', 54: 's', 55: 't', 56: 'u', 57: 'v', 58: 'w', 59:
'x', 60: 'y', 61: 'z'
}

# We know that MNIST images are 28 pixels in each dimension.
img_hieght = 22

img_width = 30
# Images are stored in one-dimensional arrays of this length.
img_size_flat = img_hieght * img_width

# Tuple with height and width of images used to reshape arrays.
```

```
img_shape = (img_hieght, img_width)

# Number of colour channels for the images: 1 channel for gray-
scale.
num_channels = 1

# Number of classes, one class for each of 10 digits.
num_classes = 62



def cvt_img2np(path):
    img = cv2.imread(path)
    return cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY).reshape(660).astype(np.float32, copy=False)

def get_images(path='./outputs'):
    images = []
    for img_path in glob(path + '/*.png'):
        images.append(cvt_img2np(img_path))
    return images



x = tf.placeholder(tf.float32, [None, img_size_flat])

y_true = tf.placeholder(tf.float32, [None, num_classes])

y_true_cls = tf.placeholder(tf.int64, [None])

weights = tf.Variable(tf.zeros([img_size_flat, num_classes]))

biases = tf.Variable(tf.zeros([num_classes]))

logits = tf.matmul(x, weights) + biases

y_pred = tf.nn.softmax(logits)

y_pred_cls = tf.argmax(y_pred, dimension=1)

cross_entropy =
tf.nn.softmax_cross_entropy_with_logits(logits=logits,

labels=y_true)

cost = tf.reduce_mean(cross_entropy)
```

```
optimizer =
tf.train.GradientDescentOptimizer(learning_rate=0.01).minimize(co
st)

correct_prediction = tf.equal(y_pred_cls, y_true_cls)

accuracy = tf.reduce_mean(tf.cast(correct_prediction,
tf.float32))

session = tf.Session()

session.run(tf.global_variables_initializer())

# Restore session
saver = tf.train.Saver()
saver.restore(sess=session,
save_path='saved_models/hcr_linear_acc_91_3')

res = session.run(y_pred_cls, feed_dict= {
    x: get_images()
    })

print(list(map(lambda x: character_mapping.get(x) , res)))
```

Result and discussion:

The most important advantage of speech over handwriting is the speed of data entry.This is because it is much easier to dictate the machine thant to write.

In the input image out of 17 characters, our model has recognized and processed 17 characters out of which 16 were correctly recognized and 1 was faulty. Hence with an accuracy of 94% for this input and an overall accuracy of 92.7%. These Results are accurate and hypothetical results that have good accuracy and clear voice file which helps us to hear after the Subject.

Conclusion

So after all these stuff.. we have a brief idea about automatic handwriting detection.

- Online and offline detection is available
- Online procedure is easier than offline procedure (difficult one)
- Today's business world need some computerized authentication for security purpose
- The AHD fulfill their need
- Handwriting recognition is important for genealogy…but it is hard
- Current methods don't work very well…and they don't operate much like the human brain

References:

G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: an extension of mnist to handwritten letters," arXiv preprint arXiv:1702.05373, 2017.

Anita Pal and Davashankar Singh,"Handwritten English Character Recognition Using Neural Network",

Manoj Sonkusare and Narendra Sahu "A SURVEY ON HANDWRITTEN CHARACTER RECOGNITION (HCR) TECHNIQUES FOR ENGLISH ALPHABETS" Advances in Vision Computing: An International Journal (AVC) Vol.3, No.1, March 2016

SANDHYA ARORA,DEBOTOSH BHATTACHARJEE,MITA NASIPURI, L.MALIK,M.KUNDU, D.K.BASU, Performance Comparison of SVM and ANN for Handwritten Devanagari Character Recognition, International Journal of Computer Science Issues (IJCSI) , Vol. 7 Issue 4, p18. (July 2010)

Responsibility of each group member :

MANIKRANTH REDDY (12) : Abstract, Introduction, Literature Review and the code

Ramprawesh kumar (9) : proposed methodology and strategies ,Result and Discussion , conclusion ,code

Pavan kumar (11) : Review and References.

Prashanth (10) : Not done anything!.