Digitizing Handwriting with a Sensor Pen:
A Writer-Independent Recognizer

Mohamad Wehbi
Machine Learning and Data Analytics Lab
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, Germany
mohamad.wehbi@fau.de

Tim Hamann
STABILO International GmbH
Heroldsberg, Germany
tim.hamann@stabilo.com

Jens Barth
STABILO International GmbH
Heroldsberg, Germany
jens.barth@stabilo.com

Bjoern Eskofier
Machine Learning and Data Analytics Lab
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, Germany
bjoern.eskofier@fau.de

Abstract—Online handwriting recognition has been studied for a long time with only few practicable results when writing on normal paper. Previous approaches using sensor-based devices encountered problems that limited the usage of the developed systems in real-world applications. This paper presents a writer-independent system that recognizes characters written on plain paper with the use of a sensor-equipped pen. This system is applicable in real-world applications and requires no user-specific training for recognition. The pen provides linear acceleration, angular velocity, magnetic field, and force applied by the user, and acts as a digitizer that transforms the analogue signals of the sensors into timeseries data while writing on regular paper. The dataset we collected with this pen consists of Latin lower-case and upper-case alphabets. We present the results of a convolutional neural network model for letter classification and show that this approach is practical and achieves promising results for writer-independent character recognition. This work aims at providing a real-time handwriting recognition system to be used for writing on normal paper.

Keywords-Sensor Pen; Online Handwriting Recognition; Convolutional Neural Network; Machine Learning

I. INTRODUCTION

Handwriting recognition (HWR) is the process of converting one's handwriting into a format that a computer recognizes. The field of HWR has gained a renewed interest over the past years due to the high demand of recognition applications for improved human-computer interaction. HWR is classified mainly into two types, offline and online recognition [1].

Offline HWR, also known as optical character recognition (OCR), takes place when only static information is available to the system. It is applied after a document is finished by the writer with the use of cameras or scanners that convert the handwriting into a digital form. This has been implemented in different application scenarios such as recognizing numbers on bank checks[2], understanding addresses on mail[3],

Figure 1. The Digipen by STABILO [5].

and reading forms[4]. Offline HWR can be useful in different domains and can be applied at any time after a document is written. However, a disadvantage of offline HWR is that it cannot be used when trying to apply recognition in real-time as a person is writing.

Online HWR (OHWR) is the analysis of spatiotemporal signals with the purpose of understanding a geometrical design. An online recognizer understands the dynamic movement of the writer's hand, which includes the writing speed and direction. It could also include different applied strokes while writing. Online HWR systems can be more accurate than offline systems since temporal information is also passed along with the spatial information, as opposed to offline systems where only static information is available. Spatiotemporal information can be helpful when trying to distinguish between similarly shaped characters by knowing the number of strokes that were needed for each character, and following different movement pauses that occur during writing when characters contain corners. In other cases, this dynamic information can confuse the system when dealing

with characters that can be written in multiple numbers of strokes which depends on the writer. In English, the average number of strokes is two strokes for uppercase letters and one stroke for lowercase letters. Cursive writing has an average of less than one stroke per letter[6].

Applications of OHWR systems vary across many domains. An OHWR system is useful for different users since it allows them to obtain the advantages of writing, with saving time by avoiding the scanning documents for digitization. Some applications are handwriting analysis in children[7], detecting alcoholism[8], and neurological analysis of patients[9].

For an OHWR system, a digitizer is required for the digitization process. Handheld devices such as tablets and smartphones recognize handwriting by following the location of the finger or stylus pen tip on the screen of the device. This has been successfully implemented in different applications but is constrained to the use of special writing surfaces. Gerth et al. [10] have shown that there are differences between writing on paper and writing on a tablet surface, where the latter requires additional movement control and presents a more challenging writing process. Therefore, to make recognition possible while writing on regular paper surfaces, devices with different sets of inertial measurement unit (IMU) sensors are required to track the movement of the writer's hand. This has been implemented in several applications that are discussed in the next section, yet the results are still not satisfactory to allow such systems to be used in real-world applications for real-time recognition systems.

When applying an OHWR system for real-world usage, different aspects should be considered for the system to be usable. One main requirement is that the system is simple to use, i.e no complex hardware with complex configuration needed. Another important aspect is that the system should work for all writers, that is, recognition should be writer-independent, and not trained for a specific writer. In this paper, we present a writer-independent system trained to recognize the Latin alphabet with the use of a sensor-enhanced ballpoint pen presented in Fig. 1. The recognizer requires no prior knowledge of the writer and can be implemented on a tablet for real-time recognition while writing on plain paper.

II. RELATED WORK

Over the past few years, several sensor-equipped devices have been designed and different approaches have been used to solve the task of real-time handwriting recognition.

In combining OCR with an IMU-based pen, Wang et al. [11] pursued the idea of using a trajectory reconstruction algorithm to estimate the position of the tip of a pen. The resulting trajectories were then classified by a regular optical digit recognition software on a tablet computer. They achieved a recognition rate of 94.6% for the ten digits.
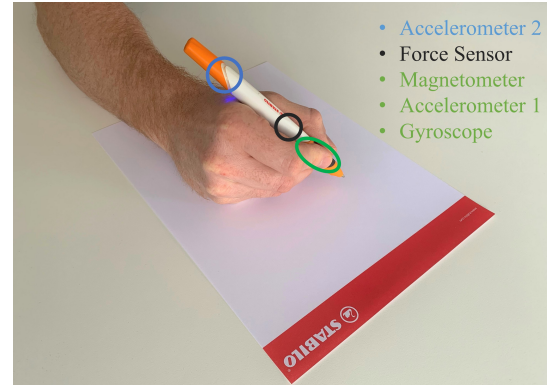

Figure 2. The location of the sensors on the Digipen [5].

Deselaers et al. [12] used gyroscope data from a mobile phone to trace an image of the hand movement while writing and used the produced image to recognize the written word following known offline writing recognition methods. They applied the system for single stroke movement of the device on the writing surface and presented the results showing the tracing accuracy of the hand movement.

Amma et al. [13] used a data glove to measure hand motion and extract gestures and characters drawn into the air. The tool was equipped with gyroscopes and accelerometers. They achieved a writer-dependent character recognition rate of 94.8% and a writer-independent rate of 81.9% using Hidden Markov Models. However, the complexity of the hardware with the process of writing in air posed a limitation for the usage of such a system as a recognizer application that can be used in regular daily situations.

Schrapel et al. [14] presented a sensor-equipped pen called Pentelligence. It senses the pen tip's motions and records the sound emissions when performing strokes. These data were used to train a Neural Network. They achieved accuracies of 78% (writer-independent) and 98% (writer-dependent) when classifying the ten digits.

Koellner et al. [15] used the Digipen by STABILO[1] to further pursue the research of online character recognition over 26 lowercase Latin letters with a collected dataset of 20,000 samples. A comparison is presented between LSTMs and other machine learning models for OHWR achieving 94% accuracy for writer-dependent classification, while showing a 52% accuracy for writer-independent classification, making the system inapplicable to be used for general recognition.

Summing up, different approaches have been taken to solve the problem of OHWR with different devices. However, various restrictions such as recognizing digits only, being user-dependent, or covering single stroke writing, and even using complex hardware limit the success of such systems in real-world applications. With such limitations,

---

[1]STABILO International GmbH (Heroldsberg, Germany)

commercial HWR applications tend to apply OCR methods on images produced by tracing a pen trajectory over a specific writing surface. Thus, we present a system using a sensor pen input device for online character recognition which, in contrast to the related work, relies on a pen without the need for additional hardware, and based on the research done, we provide better results over the Latin alphabet for writer-independent recognition.

III. METHODOLOGY

A. Digipen

The approach presented in this paper uses the STABILO Digipen, which is a sensor-enhanced ballpoint pen equipped with five sensors: Two accelerometers, a gyroscope, a magnetometer, and a force sensor distributed over the pen as shown in Fig. 2. The accelerometers were adjusted to a range of $\pm 2g$ with a resolution of 16 Bit (front) and 14 Bit (rear), the gyroscope has a range of $\pm 1000\,°$/s with a resolution of 16 Bit, the magnetometer has a range of $2.4\,$mT with a resolution of 14 Bit and the force sensor has a measurement range of 0 to $5.32\,$N with a resolution of 12 Bit. The pen is also provided with a soft-touch grip zone for holding the pen, and includes a Bluetooth module that enables the live streaming of data to a handheld tablet or smart phone. The data provided by the recordings were used in its raw form without the need for further calibration.

B. Data Acquisition

The approach presented in this paper utilizes the pen's raw sensor data to classify letters. For data collection, a mobile app was provided by STABILO for saving the data stream while using the sensor-equipped Digipen as the input device. The pen is connected to the app via Bluetooth Low Energy and the sensor data are transmitted at a rate of 100 Hz throughout the session.

To carry out the data collection process, several data recording sessions were conducted with 114 right-handed subjects who volunteered to participate. The subjects were requested to insert data such as name, age, gender, and dominant hand before the recording starts. The app displays the letter to write, as displayed in Fig. 3, going through the uppercase then lowercase alphabet and repeats this process up to 6 times. At the end of the recording, two files were saved on the tablet, the data stream file of all the sensor values throughout the complete recording session, and the labels file in which the ground truths are identified based on the time when they were displayed on the screen. These recordings resulted in approximately 14,870 uppercase and 14,841 lowercase letter recordings since not all subjects fully completed the recordings.

For the data acquisition process, some boundary conditions had to be met: Only right-handed persons over the age of 16 were allowed to participate in the recordings. The volunteers were instructed to hold the pen in its correct
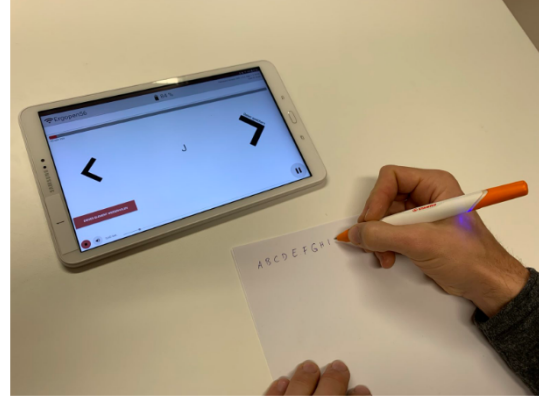

Figure 3. The data recording app provided by STABILO [5].

orientation (with the logo facing upward). Furthermore, the writing surface had to be horizontal and padded by at least 5 paper sheets to create homogeneous conditions throughout different recording sessions.

C. Data Preparation

Following the data collection process, the labels file of each recording session was used to split the sensor data file into single letters with the relative timeseries data. Each split single-letter file consists of the sensor data that were transmitted while the letter was shown in the recording app. This includes phases where the pen was hovering before and after writing the letter.

To train only on meaningful data, each sample was filtered keeping only the timesteps when the user was writing and not hovering between two different letters. This was applied by removing all datapoints of a sample with a force sensor value less than 0.2 N, before and after writing the letter, while keeping the low values in between the start and end of a letter, to represent multiple strokes. Fig. 4 shows a force sensor recording of the letter 'B', where the first peak in the signal represents the writing of the vertical line, and the other peaks representing the writing of its two semi-circles.

However, removing the datapoints with low force signals resulted in the samples having different lengths, which was mainly due to different writing speeds between different subjects, or between different letters of a single subject. Thus a resampling technique was required to equalize the length of the samples.

Each letter sample was linearly interpolated to a length of 50 timesteps, which represented the mean length of all the data samples collected. The interpolation allowed an easier fitting of the data into the model and was assumed to regularize the pace of writing that helped the model to better learn and classify the data. Samples with a length outside the interval of 10-500 timesteps were considered as outliers and removed from the dataset. Finally, each sample was normalized using L2 feature-wise normalization which
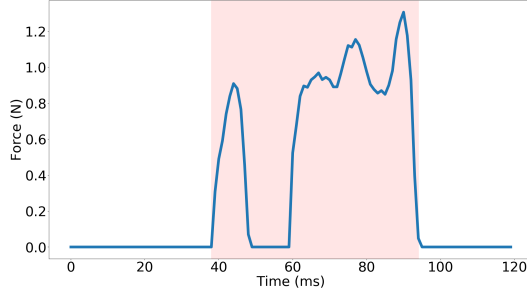
Figure 4. A graph showing the raw force sensor values of a single letter with the highlighted part of the recording to be used as a single input sample for the neural network.

allowed the model learning process to run more smoothly and provided better results.

D. Network Architecture & Training

Previous works with similar data types mainly used a Recurrent Neural Network (RNN) model since RNNs were designed to work with sequential data to retain and remember information throughout a long sequence, and are also easier to use when dealing with data of unknown lengths. However, LSTMs performed poorly when dealing with writer-independent recognition, which is mainly because a sequence of the same letter changes from one writer to another.

The model used in this paper was based on a 1D-Convolutional Neural Network (CNN). The data samples were of fixed lengths after the interpolation, which was necessary to fit the data into a CNN. The usage of a convolutional model was due to the ability of CNNs to capture local dependencies and scale-invariant features of signals[16]. The convolutional model was able to extract important features without the need for any further preprocessing of the data.

The model was implemented with the Keras library[17]. It consisted of a 13 channel input layer (12 channels for the x,y,z axis of each of the four tri-axial sensors and one channel for the force sensor), followed by three hidden layers, with an output layer of 26 classes. The training of the network was done over 50 epochs. The 'Relu' activation function was used for all hidden layers, with a 'Softmax' activation at the output layer. BatchNormalization was done after all hidden layers, and dropout was applied at the first and second hidden layers with a 40% dropout rate. As a loss function, the 'categorical crossentropy' loss was used, and 'Adam' optimizer with a learning rate of 0.001. The convolutional layers consisted of 64 filters each, with a kernel size of 4, and the fully connected hidden layer consisted of 100 units. The same model configuration was used for both uppercase and lowercase letter classification. Fig. 5 shows a summary of the architecture used in the model.
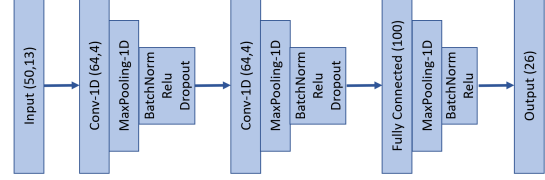

Figure 5. A summary of the network architecture used.

To better provide the reader with a comparison to previous works, another model was implemented with an LSTM architecture with two hidden layers using the 'Relu' activation function. The other hyperparameters of the network and the training were left unchanged. This was done to show how the different parts of our experiments contributed to providing a better recognition accuracy.

IV. RESULTS

We show the results obtained by using the data collected with the STABILO Digipen by the CNN and the LSTM models. This work focuses primarily on the writer-independent character recognition since writer-dependent is in most cases not applicable for a general real-world use-case scenario. However, since in some commercial application scenarios, it is common to have the user adapt the pre-trained model with some individual training data to enable personalization, we run experiments over a writer-dependent recognition scenario and include the results.

The complete dataset was used in both scenarios. For the writer-independent case, we exclude a number of subjects from the training dataset for testing, while in the writer-dependent case, a writer that contributed to the training set can also contribute samples to the test set.

The models were trained for the uppercase and lowercase classification separately. The accuracy was the metric used for the evaluation. Since the writing shape variance for the uppercase letters is lower than that in the lowercase letters, the accuracy was assumed to be higher in the former case. The results of each of the two scenarios are discussed separately.

A. Writer-dependent case

A system that has seen samples of how a specific user writes is suspected to find it easier to recognize other samples of this specific user, since each individual has a certain way of holding the pen, moving the pen, writing letters in a specific way, and specific writing speed. So, having learned beforehand how a writer writes, a model can use these qualities for better recognition of what is written. Given also how that dataset was recorded, having randomly split the data into training and test sets, a specific sample cannot be in both sets, but it is likely that another sample of the same person would be available in the test set since every subject recorded the complete alphabet more than once.