End-to-End ECG Digitisation with Soft Segmentation and BiLSTM Signal Reconstruction

Adrian McIntosh[1], Mojtaba Lashgari[1], Longa Maimbo[2], Abhirup Banerjee[1]
[1] Institute of Biomedical Engineering, Department of Engineering Science, University of Oxford, UK
[2] Department of Electrical Engineering, University of Cape Town, South Africa

Abstract

ECG is a widely used and accessible tool for cardiac diagnosis and monitoring, generating vast amounts of data each year with strong potential for AI-driven population studies. However, much of this data remains in paper format, limiting computational use. Digitising these records, especially via images, is challenging due to uneven illumination, noise, and artefacts, such as handwritten annotations that can obscure the signal. In this paper, we propose a fully automated pipeline for digitising paper ECG signals on the PhysioNet 2025 ECG Digitisation Challenge dataset. The proposed method consists of three stages: image rectification, probability map segmentation, and a Bidirectional Long Short-Term Memory (BiLSTM)-based signal reader. Unlike conventional approaches that utilise rule-based trace-following methods, the proposed method converts segmentation outputs using sequential modelling, enabling robust reconstruction in the presence of realistic input distortions and artefacts. Evaluated end-to-end on a held-out test set, the pipeline achieved a mean signal-to-noise ratio of 16.89 $\pm$ 3.11 dB, demonstrating the potential of a fully automated ECG digitisation framework.

1. Introduction

Electrocardiography (ECG) remains one of the most accessible and widely used tools for diagnosing and monitoring cardiac function. Each year, more than 200 million ECG recordings are generated worldwide, presenting significant opportunities for large-scale population studies using advanced artificial intelligence (AI) models [1]. However, a major barrier to realising this potential is the limited availability of digital ECG data. Many clinical systems still produce ECGs as paper printouts, which are not readily amenable to computational analysis. Converting these records into digital formats introduces additional challenges, particularly when using image-based approaches. Issues such as uneven illumination, noise, and artefacts, including handwritten annotations or markings, can obscure the ECG signal and degrade the quality of the data [2]. In addition, digital ECGs serve several important roles in clinical environments. They facilitate efficient storage, retrieval, and sharing of patient data across healthcare systems, improving workflow and continuity of care. Recovering time-series signals from real images of printed graphs involves several challenging problems, including geometric distortion, stains, glare, overlapping traces and varying image resolutions [2].

Studies addressing ECG digitisation generally follow a similar three-stage workflow [2]: 1) image rectification, 2) ECG segmentation, and 3) conversion of ECG segmentation to time series. For the first stage, several rectification methods have been used in previous works, such as the 'Gridder Algorithm' [2], frequency-domain based rotation [3], and deep neural network based methods [4]. During the segmentation stage, ideal conditions allow some methods to use simple HSV colour filters to separate the gridlines from the waveform part of the image [5], but in cases with additional artefacts and distortions, U-Net variants are commonly used to generate binary segmentation masks [2–4]. In some cases, YOLO was additionally used to detect and separate bounding boxes for each lead to work around non-standard signals and machine errors [3, 4]. The third stage, converting the segmentation mask into a time-series ECG, is typically achieved using classical or rule-based methods to infer the waveform from the segmentation output [6]. A primary drawback of many studies lies in the third stage, where limited flexibility in handling diverse signal variations can lead to significant errors when applied to real-world ECG images.

In this paper, we follow the same three-stage workflow. However, to address the gap associated with the third stage, we propose a fully automated pipeline that deals with a variety of input distortions and image acquisition formats (scans, mobile phone photos, etc.) available on the PhysioNet 2025 ECG Digitisation Challenge dataset. Specifically, we introduce a novel attention-masked Bidirectional Long Short-Term Memory (BiLSTM) module for signal conversion, a more general alternative to rule-based post-segmentation methods. Our approach demonstrates strong fidelity on synthetic data and is designed to be robust to distortions and artefacts in ECG digitisation.

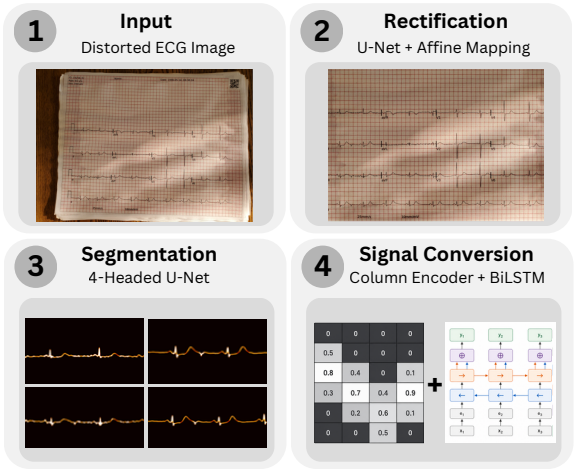

Figure 1: An overview of the proposed digitisation pipeline.

2. Methods

2.1. Dataset and Experimental Setup

The 2025 Physionet ECG Digitisation Challenge Dataset was used for this work [7], consisting of 977 12-lead ECG signals, originating from the PTB-XL dataset [8]. For each signal, ECG-Image-Kit [9] was used to generate 9 synthetic images of the analogue recording. These synthetic images include views in different forms, such as mobile phone images and scans, with various artefacts including stains, folds, and glare. Each signal includes the ground-truth time-series signal measured in millivolts, recorded at sampling frequencies between 2,500 and 10,250 Hz.

Since the dataset includes multiple synthetically distorted images for each ECG signal, the dataset is split such that each ECG only appears in one split. These splits include 70% training, 15% validation, and 15% test sets. The held-out test set consists of 147 randomly selected ECG signals, each with nine corresponding synthetic images, resulting in 1,323 image-to-signal conversions.

Both the segmentation model and the BiLSTM model are trained for 100 epochs, where each epoch samples one randomly selected image per ECG signal. Signal quality is evaluated using the PhysioNet competition signal-to-noise ratio (SNR) metric, which shifts the resulting signal by up to ± 0.2s and takes the best score, removes per-lead DC offset, and computes SNR in dBs from the ratio of summation of signal power to summation of noise power across all 12 leads.

2.2. Model Overview

This subsection provides an overview of the proposed model, which is composed of three main stages:

1. **Rectification:** Images are first cropped and projected onto a standard geometric plane, using a pretrained U-Net and affine transformations.
2. **Segmentation:** A second U-Net architecture is trained and applied to generate a probability map indicating the likelihood that each particular pixel belongs to the waveform rather than background.
3. **Signal Conversion:** An integrated BiLSTM model is used to a) localise the precise ECG position within the probability map and b) reconstruct the time-series signal at the desired temporal resolution through interpolation.

Figure 1 provides a visual overview of the process.

2.2.1. Rectification

The rectification stage aims to remove background grid patterns (e.g., red grids) and correct geometric distortions in the ECG image. A pre-trained U-Net-based model, adapted from [10], is first applied to approximately predict a grid-intersection probability map. Next, a connected component analysis is performed on these predictions to extract the grid candidates. Finally, the reconstructed grid is converted into a dense sampling map, and the image is warped onto a fixed reference frame, producing a rectified output.

2.2.2. Segmentation

The segmentation stage generates probability maps indicating the location of ECG signals on the rectified image produced in the previous stage. A U-Net is trained to map the rectified input to a four-channel output, with each channel corresponding to one ECG trace. Rather than binary masks, the targets are defined as probability maps centred on the signal trajectory, allowing the model to better handle varying line thickness and intersections. The training dataset is generated by projecting the ground-truth time-series onto the image space and applying a Gaussian blur to capture the full signal width, resulting in probability maps that represent the likelihood of signal presence at each pixel.

Standard segmentation losses (e.g., binary cross-entropy or Dice) are not well-suited to soft, Gaussian-like targets. Instead, we define a regression loss composed of four terms. Let $\hat{p} = \sigma(z)$ denote the sigmoid output and $p \in [0, 1]$ the target probability map. To emphasise the signal region, a weighting term $w = 1 + (\lambda_s - 1)p$ is introduced, with signal-emphasis factor $\lambda_s = 6.0$. Two weighted pixel-wise regression losses are then applied: a mean squared error (MSE) loss

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N} \sum_{n} (\hat{p}_n - p_n)^2 \, w_n \tag{1}$$

and an ℓ_1 loss

$$\mathcal{L}_{\ell_1} = \frac{1}{N} \sum_{n} |\hat{p}_n - p_n| \, w_n \tag{2}$$

where the MSE term emphasises large local deviations in the predicted heatmap, while the ℓ_1 term improves robustness to small but systematic centreline offsets along the ECG trace.

To further prioritise accuracy near the signal centreline, a peak-focused term is introduced:

$$\mathcal{L}_{\text{peak}} = \frac{1}{N} \sum_{n} (\hat{p}_n - p_n)^2 \, p_n^2. \tag{3}$$

Since each output channel represents one line of the ECG image, we use an inter-channel penalty to discourage overlap between them, defined as:

$$\mathcal{L}_{\text{overlap}} = \frac{1}{\binom{C}{2}} \sum_{i<j} \frac{1}{N} \sum_{n} \hat{p}_n^{(i)} \, \hat{p}_n^{(j)}. \tag{4}$$

The total loss is given by

$$\mathcal{L} = \lambda_{\text{MSE}} \, \mathcal{L}_{\text{MSE}} + \lambda_{\ell_1} \, \mathcal{L}_{\ell_1} + \lambda_{\text{peak}} \, \mathcal{L}_{\text{peak}} + \lambda_{\text{overlap}} \, \mathcal{L}_{\text{overlap}} \tag{5}$$

with experimentally measured weights $\lambda_{\text{MSE}} = 1.0$, $\lambda_{\ell_1} = 0.5$, $\lambda_{\text{peak}} = 2.0$, and $\lambda_{\text{overlap}} = 0.02$.

2.2.3. Signal Conversion

Converting a probability map into a time-series ECG requires extracting the ECGs from a 2D probability map, along with transforming the image's temporal resolution to a fixed time-series temporal resolution. We addressed these issues through three stages:

Stage 1: Per-column encoding. Given a probability map $\mathbf{P} \in [\mathbf{0}, \mathbf{1}]^{\mathbf{H} \times \mathbf{W}}$, each column c is encoded into a vector $\mathbf{z_c} \in \mathbb{R}$, providing a sequence $\mathbf{Z} = (\mathbf{z_1}, \ldots, \mathbf{z_W})$ of length equal to the image width W. For each column, the k highest-probability pixels are retained, producing pairs $(a_{c,i}, \, p_{c,i})$ for $i = 1, \ldots, k$ where $p_{c,i}$ is the probability of the i-th retained pixel, and $a_{c,i} = r_{\text{base}} - r_{c,i}$ is the signed difference of the given pixel from the baseline row r_{base}, positive above the baseline. Columns with fewer than k non-zero probabilities are padded with $(0, 0)$, and a corresponding binary mask $m_{c,i} \in \{0, 1\}$ indicates which entries are real.

Each candidate is passed through a shared two-layer multilayer perceptron (MLP), $\mathbf{e_{c,i}} = \text{MLP}(\text{a}_{\text{c,i}}, \text{p}_{\text{c,i}}) \in \mathbb{R}$ and scored against a learned vector $\mathbf{q} \in \mathbb{R}$:

$$\alpha_{c,i} = \frac{\exp(\mathbf{q}^\top \mathbf{e_{c,i}}) \, \mathbf{m_{c,i}}}{\sum_{j=1}^{k} \exp(\mathbf{q}^\top \mathbf{e_{c,j}}) \, \mathbf{m_{c,j}}}, \qquad \mathbf{z_c} = \sum_{\mathbf{i=1}}^{\mathbf{k}} \alpha_{\mathbf{c,i}} \, \mathbf{e_{c,i}}. \tag{6}$$

Stage 2: BiLSTM. The sequence $\mathbf{Z}$ is processed by a BiLSTM [11]. At each position c, a forward LSTM reads left-to-right and a backward LSTM reads right-to-left, giving hidden states $\overrightarrow{\mathbf{h}}_c, \; \overleftarrow{\mathbf{h}}_c \in \mathbb{R}'$ respectively. Their concatenation

$$\mathbf{u_c} = [\, \overrightarrow{\mathbf{h}}_\mathbf{c}; \; \overleftarrow{\mathbf{h}}_\mathbf{c} \,] \tag{7}$$

captures context from both neighbouring columns and from columns far away in either direction. A linear layer maps each $\mathbf{u_c}$ to a scalar value s_c, the predicted offset of the trace from r_{base} in pixels.

Stage 3: Upsampling. These predictions $\mathbf{s} = (s_1, \ldots, s_W)$ at the image width resolution are upsampled by linear interpolation to the ground-truth sampling rate $\hat{\mathbf{y}} = \text{Interp}(\mathbf{s}) \in \mathbb{R}^{\mathbb{T}}$, where $T > W$ is the number of samples in the ground-truth signal.

Loss. Training minimises a smooth-ℓ_1 and a total-variation penalty loss:

$$\mathcal{L} = \frac{1}{T} \sum_{t=1}^{T} \ell_{\text{s1}}(\hat{y}_t - y_t) \; + \; \lambda \frac{1}{T-1} \sum_{t=1}^{T-1} |\hat{y}_{t+1} - \hat{y}_t|, \tag{8}$$

where T is the number of samples (or time steps) in the ground-truth signal, $\hat{y}_t$ is the predicted signal value at time step t, y_t is the ground-truth signal value at time step t, λ is the regularisation weight controlling the trade-off between the two terms, and the smooth-ℓ_1 loss, ℓ_{s1}, is:

$$\ell_{\text{s1}}(x) = \begin{cases} \frac{1}{2} x^2 & \text{if } |x| < 1, \\ |x| - \frac{1}{2} & \text{otherwise.} \end{cases} \tag{9}$$

The ℓ_{s1} term is less sensitive to the large transient errors near QRS peaks than standard MSE, and the total variation term suppresses high-frequency noise in the predicted signal.

At the end of this stage, each line is split into the individual 12 leads, by dividing the number of samples in each line into four segments. Additionally, the continuous predicted pixel values are converted to mV values using the standard scaling for the dataset.

3. Results

In this section, we summarise the performance of the proposed pipeline. Although the method consists of three separate stages, evaluation is conducted end-to-end using SNR on a held-out test set.

The method achieved an average SNR of 16.89 $\pm$ 3.11 dB. Figure 2 shows the box-and-whisker plots across different distortion types. A representative example (with an SNR score close to the average) input, segmentation, and output is shown in Figure 3.

4. Discussion and Conclusion

The proposed pipeline achieved SNR of 16.89 ± 3.11 dB on the held-out test set. Notably, performance remained consistent across varying image quality and distortion types.

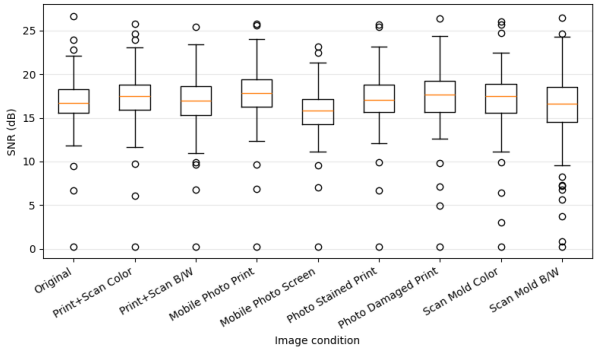

Figure 2: Box-and-whisker plots for synthetic test images based on the distortions applied to them.

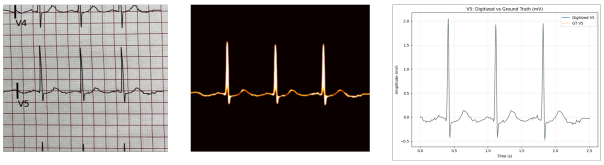

Figure 3: An example input (left), segmentation (middle), and digitised output (right) applying to the full pipeline. The SNR is 17.06 dB.

As shown in Figure 2, most of the results fall within an acceptable range regardless of the type of distortion applied, indicating robustness to different acquisition conditions and suggesting potential for real-world applicability.

A primary strength of the approach lies in the combination of probability map segmentation and BiLSTM-based signal conversion. Rather than relying on rule-based methods for segmentation to signal conversion, the model learns to map a probabilistic pixel representation directly to a continuous signal. By leveraging sequential context, it can resolve ambiguities caused by image artefacts such as stains and glare, producing smooth and plausible waveforms.

While the majority of cases are successful, the remaining errors are primarily observed in regions where the signal is heavily obscured or locally ambiguous due to overlapping ECGs. Additionally, both training and evaluation were conducted on a synthetic dataset, so performance on real scanned or photographed ECGs remains to be validated.

Overall, the results demonstrate the potential of the proposed pipeline for ECG digitisation in practical settings, including scenarios involving mobile phone images, with artefacts and geometric distortions. Future work will focus on evaluating the method on real-world ECGs across a wider range of layouts and acquisition conditions.

Acknowledgments

The authors acknowledge the use of the services/facilities of the Institute of Biomedical Engineering (IBME), Department of Engineering Science, University of Oxford. The authors express no conflict of interest. The work of AB was supported by the Royal Society University Research Fellowship (grant no. URF\R1\221314). The work of ML was supported by the Royal Society Enhanced Research Expenses Grant, awarded to AB. AM is supported by FirstRand Laurie Dippenaar Scholarship.

References

[1] Rafie N, et al. ECG Interpretation: Clinical Relevance, Challenges, and Advances. Hearts 2021;2(4):505–513.

[2] Demolder A, et al. High precision ECG digitization using artificial intelligence. Journal of Electrocardiology 2025; 90:153900.

[3] Silva R, et al. YOUR-Lead: YOLO and U-Net for Reconstruction of ECG Lead Signals. In Computing in Cardiology Conference (CinC), volume 51. 2024; 1–4.

[4] A. Verlyck M, et al. WAVIE: A Modular and Open-Source Python Implementation for Fully Automated Digitisation of Paper Electrocardiograms. In Computing in Cardiology Conference (CinC), volume 51. 2024; 1–4.

[5] Wu H, et al. A fully-automated paper ECG digitisation algorithm using deep learning. Scientific Reports 2022; 12(1):20963.

[6] H. Krones F, et al. Combining Hough Transform and Deep Learning Approaches to Reconstruct ECG Signals From Printouts. In Computing in Cardiology Conference (CinC), volume 51. 2024; 1–4.

[7] Reyna MA, et al. Physionet - digitization of ecg images. https://kaggle.com/competitions/physionet-ecg-image-digitization, 2025. Kaggle.

[8] Mehdi NA, Drigh AA. PTB-XL-Image-17K: A Large-Scale Synthetic ECG Image Dataset with Comprehensive Ground Truth for Deep Learning-Based Digitization. arXiv preprint arXiv260207446 2026;1 – 9.

[9] Shivashankara KK, et al. ECG-Image-Kit: a synthetic image generation toolbox to facilitate deep learning-based electrocardiogram digitization. Physiological Measurement 2024; 45(5):055019.

[10] Ahmed K. hengck23-submit-physionet. Kaggle dataset. [Online]. Available: https://www.kaggle.com/datasets/kami1976/hengck23-submit-physionet, 2026.

[11] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation 1997;9(8):1735–1780.

Address for correspondence:

Adrian McIntosh
Institute of Biomedical Engineering, Department of Engineering Science, University of Oxford, Oxford, OX3 7DQ, UK
adrian.mcintosh@eng.ox.ac.uk