An Open-Source Wearable Sensor System for Detecting Extravasation of Intravenous Infusion

Huang-Chen Lee, *Senior Member*, Jheng-Sing Lin

Abstract—Extravasation is a common hazard in intravenous therapy. However, in clinical practice, the nursing staff is responsible for checking the injection status regularly, which leads to the risk of delayed treatment of the extravasation of intravenous therapy. In this study, we propose an open-source solution to detect the early signs of extravasation—including the wearable sensor to collect physical signals from the skin, the simulation platform to simulate extravasation on artificial skin, and the data server to collect and analyze data of occurring extravasation. Pressure, body temperature, and optical sensors were integrated into the wearable sensor and evaluated to understand their effectiveness in detection. We also propose the Light-ConvLSTM model can predict extravasation with a comprehensive evaluation and show the advantages and feasibility of the proposed concept. The results show the proposed design with Light-ConvLSTM can achieve significant performance: the extravasation detection rate can be raised to 83.7%, while the false alarm rate is only 6.2%.

***Index Terms*—**Embedded system, system integration, extravasation detection, sensor design, bio-sensor, simulation platform, medical device, deep learning.

I. Introduction

Extravasation is the leakage of intravenous (IV) fluid out of a vein into the surrounding tissues, as Fig. 1 shows. The effects of extravasation can be catastrophic and can include pain, swelling, and even tissue hardening and wound ulcers. When the signs of extravasation occur, to reduce injury, the injection must stop immediately, and the corresponding treatment must be initiated. However, due to the long course of IV infusion, it is almost impossible for the nursing staff to monitor the infusion status throughout the whole process. Moreover, the early signs of extravasation are difficult to recognize with the naked eyes. All these factors expose IV infusion therapy to the risk of delayed treatment.

Recent studies have started to use nonintrusive sensors to monitor the physical changes of the skin near the catheter tip to detect early signs of extravasation. Cheng et al.'s design [1] indicates extravasation can cause skin swelling that can be observed by sight. Therefore, they proposed the use of a specially designed strain gauge, which is in the form of an adhesive firm near the catheter tip to detect skin change during IV infusion therapy. The electrodes of the strain gauge are pushed by the swelling skin, causing varying resistance, to detect the occurring extravasation. Another study [2] takes a different approach by observing the temperature distribution and change with a thermographic camera. The researchers noticed skin temperature drops below room temperature if extravasation is occurring and the use of the temperature change to determine whether extravasation is possible. Near-infrared spectroscopy (NIRS) is also been used to detect extravasation [3] by observing the different wavelengths of light been absorbed by tissues, while similar technology is already been used in measuring blood oxygen, blood glucose level, and heart rate. However, different sensors detect different types of skin physiological changes but may still suffer from the noise of sensor reading or other environmental interferences. A combination of multiple types of sensors is a straightforward approach to overcome these issues. A study [4] integrated the use of an electrocardiogram and skin pressure sensor to detect extravasation. This study determined the occurrence of extravasation using a simple voting mechanism; if both sensors detect the signal of extravasation over the predefined threshold, then the system issues a warning message.

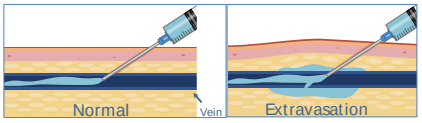

Fig. 1. Intravenous therapy under normal conditions (left) and extravasation (right)

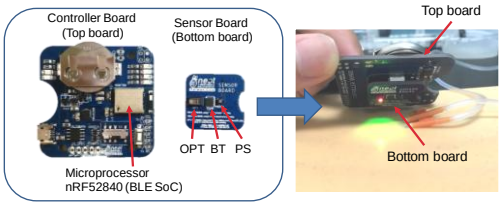

Fig. 2. The proposed wearable sensor board for detecting extravasation

While previous studies have explored the use of different sensors and design approaches to accurately determine extravasation, they have mostly involved sensor system designs and have not mentioned how to analyze the collected data and detect the pertinent phenomenon. There is a desperate need for a general IV infusion monitoring system with the characteristics of reliable operation in a hospital.

In this study, we propose an intelligent wearable sensor system that can accurately determine extravasation. This system contains a wearable sensor board (as shown in Fig. 2) that is assembled using the top and bottom boards, which can attach to the skin near the catheter tip. The sensor board can collect physical changes of the skin—including color, deformation, and temperature—and the collected data can be transferred to a server to determine the possibility of extravasation and then issue alarms. As the data collected is disorderly and difficult to understand, this study applies deep learning techniques, including the use of a convolutional neural network (CNN) and long short-term memory (LSTM), to analyze the collected data and determine the possibility of extravasation. The results of the experiments show the

proposed system can, to a high degree, accurately detect extravasation while keeping the false alarm rate low. This paper is based on our previous work [17].

The goal of this study is the collection of physical changes during IV therapy and early detection of extravasation. The contributions of this study are

1. To our knowledge, no previous studies have discussed the design of a wearable sensor system that can detect extravasation of IV infusion using a deep learning algorithm, with comprehensive evaluations. We offer the first sensor system with an evaluation platform that ensures extravasation can be determined with high accuracy while keeping the false alarm rate low.
2. The data of skin changes—including color, swelling and temperature changes—collected by the system are fused and analyzed using a deep learning algorithm. By combining multiple sensors' data, the proposed design is able to deal with the data from some ineffective sensors while compensating for the other sensors to increase performance. Extravasation can be detected more accurately, and early warnings can be given take immediate action.
3. The resource required to make predictions was analyzed, ensuring model inference can be locally executed on the microprocessor without transmitting raw data to a remote server. This approach can allow the sensors detect the occurrence of extravasation while saving valuable energy for wireless data transmission and conserving the patient's privacy, as the raw data are stored in the device.

The rest of this paper is organized as follows. Section 2 discusses related works and our goals. Section 3 describes the details of the system's architecture and design. Section 4 presents the experiment and evaluation of our design. Section 5 shares our conclusions.

II. Related Works and Design Goals

Several works have attempted to design special sensors and systems to detect the occurrence of extravasation during IV therapy. As extravasation is caused by the leakage of fluids from a vein to tissues during IV therapy, it is usually noticed late, when a large swelling or bump is visible.

Cheng et al. [5] designed a special strain gauge sensor to measure swelling and detect extravasation. The sensor was attached to an adhesive film and pasted on the patient's skin to detect swelling at the IV cannula site. The proposed design can measure the bump height and the pressure and output the corresponding resistance values. The sensor was also evaluated in an ex vivo experiment, showing the infusion fluid volume and bump height were highly correlated to the output resistance values; however, this paper focused more on the pressure sensor design but lacked a detailed analysis of the data collected.

Another study [6] used thermography to identify the occurrence of extravasation during IV therapy. As the IV liquid temperature is lower than the body's temperature, a lower-temperature region from the infusion site margin can be observed while extravasation is occurring. This study demonstrated thermography can be used to detect extravasation, with reasonable accuracy. The data analysis of this work showed the temperature difference can be used to infer the occurrence of extravasation, but no concrete algorithm was proposed to help systematically detect the phenomenon of interest. Meanwhile, the air temperature of the surrounding environment might also potentially cause the temperature change of the patient's skin and result in a false alarm.

NIRS has also been evaluated to detect extravasation [7], as the tissue absorption of 760nm IR is changed, while the fluid volume in the tissue varies. The study showed the proposed NIRS sensor can detect extravasation at a sensitivity of up to 0.3 ml at a subcutaneous depth of 2 cm. This study opened up a new approach to detect extravasation; no detailed data analysis and algorithm were discussed to validate the performance of this method. A commercial product [18] had been evaluated [19] to detect extravasation based on NIRS. Since there is no detailed information on how this product detects and analyzes the collected data to detect extravasation, we cannot compare it with our design directly. However, we will evaluate the performance of detecting extravasation with only NIRS in the evaluation section, which shows our platform with multiple sensors might have better accuracy and lower false alarm compare to the solution with a single NIRS sensor.

Another study [8] combined two sensing modalities, electrical bioimpedance and skin strain sensing, for detecting extravasation. The authors proposed a simple fusion rule—that is, the logical AND, logical OR, and majority rule—to increase the accuracy of output results. This study demonstrated the sensor-fusion approach can help to reduce the false alarm rate. Nevertheless, the decision model was too simple and might be unpractical in real usage if multiple noise sources are involved.

Table 1 Comparison of the related works

Name of Study	Type of sensor	Algorithm proposed	Open-Source
Cheng et al. [5]	Strain gauge	N/A. Focused on sensor design.	No
Y. Matsui et al [6]	Thermography	No concrete algorithm was proposed.	No
Y.-C. Du et al[7]	Near-infrared (NIRS)	N/A. Focused on sensor design.	No
ivWatch [18]	Visible light and near-infrared (NIRS)	N/A as it is a commercial product.	No
A. O. Bicen et al [8]	Bioimpedance and skin strain	Only simple AND-OR logic	No
The proposed study	Pressure, body temperature and optical sensor (red, infrared, and green light)	Deep learning algorithm: Light-ConvLSTM	Open-Source

As the comparison shown in the Table 1, some of the previous studies have majorly focused on exploring new types of sensors to detect extravasation, but not from the perspective of designing a system. Only the ivWatch is a complete system that uses visible and near-infrared light to determine extravasation; however, as a commercial product, the detailed specifications of the sensors are unavailable publicly. In contrast to the previous studies, this paper proposes a complete, open-source extravasation detection system—including the data acquisition, data processing, and deep learning model—to increase extravasation detection rates and lower false positive alarm rates. The design goals of the proposed system are as follows:

1. The wearable sensor must be able to collect multiple physical features during IV therapy with different types of

sensors; avoiding the noise signal of a single sensor can cause false alarms—for example, the skin's temperature being changed by an air conditioner or cooling fan. The proposed system must be able to detect extravasation with higher confidence while lowering the false alarm rate. This is very important to attract the nursing staff to use this system.

2. The raw data collected from multiple sensors must be processed and fused with a proper deep learning algorithm to raise the success rate of detecting extravasation.
3. For power saving and privacy concerns, the system must have the flexibility that can inference deep learning model and make predictions using a memory-limited microprocessor. Therefore, the design of the deep learning model must consider the resource limitation of microprocessors.
4. The design should be empirically verified on a simulation platform to ensure the system can work in practical scenarios. The simulation platform must be able to repeat the phenomenon of extravasation in IV therapy to create sufficient data for analysis. Human intervention must be minimized during the experiments on the simulation platform.

These goals were converted into system requirements for the design of the system, which is described in the next section.

III. System Architecture and Design

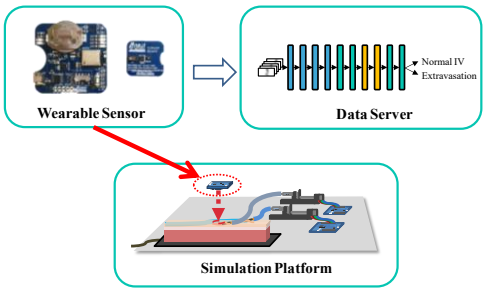

Fig. 3. The proposed system architecture

In this section, we describe the design of the system. The proposed system consists of three parts, as shown in Fig. 3.

1. **Wearable sensor:** The wearable device integrated multiple sensors to sense the skin's symptoms when extravasation occurs.
2. **IV simulation platform:** The IV simulation platform simulates normal and abnormal (i.e., extravasation) IV procedures with artificial skin for the wearable sensor to collect data.
3. **Data server:** The data server analyzes the collected data from the wearable sensor, trains and inferences model, outputs the probability of occurring extravasation.

We introduce the three parts in the following subsections.

3.1 Wearable Sensor

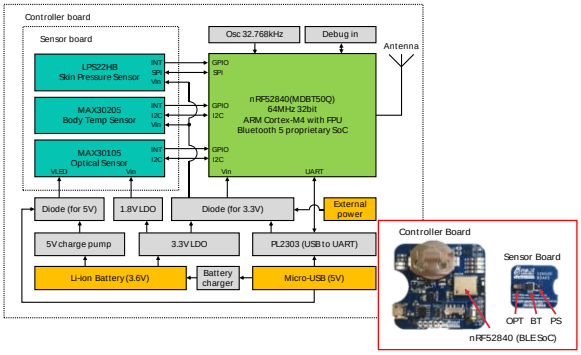

Fig. 4. The hardware architecture of the wearable sensor

As mentioned in the previous section, some symptoms can be referred to as the indicators of occurring extravasation. Therefore, the wearable sensor is designed to measure the symptoms and collect data during IV therapy. As illustrated in Fig. 4, the proposed wearable sensor customized by the authors is a 4.5 cm by 4.5 cm PCB board based on a Nordic nRF52840[10] ARM Cortex-M4 microprocessor with Bluetooth wireless communication capability and can be powered by an internal coin battery or external USB power. This microprocessor has 1 MB Flash and 256 KB RAM for storing program and data. This wearable device integrates three sensors—including the skin pressure sensor (STMicroelectronics LPS22HB) [11], the body temperature sensor (Maxim MAX30205) [12], and the integrated optical sensor (Maxim MAX30105)[13]. The sensors will be noted by **PS** (Pressure sensor), **BT** (Body temperature sensor), and **OPT** (Optical sensor), respectively, for short. These sensors are designated to detect the three symptoms of extravasation:

1. **Swelling of the skin** (refer to [5], [8]): This study uses a MEMS skin pressure sensor (PS; LPS22HB), which is a high-accuracy piezoresistive pressure sensor in a small form-factor that can be attached to the PCB. LPS22HB integrates an analog-to-digital converter and can output data via SPI or I2C to a microprocessor.
2. **Skin temperature changes** (refer to [6]): This wearable sensor integrates a BT (MAX30205), which is specifically designed to measure skin temperature changes. It can measure temperature at a resolution of 0.1℃ in the range of 37℃ to 39℃ and outputs digital data via I2C to a microprocessor while consuming an extremely low operating current (600 μA).
3. **Abnormal changes in fluid volume in tissues** (refer to [7]): The OPT (MAX30105)—which is composed of red, infrared, and green light LEDs and a photodiode—is integrated into the wearable sensor to detect fluid volume in tissues. The LEDs emit different lights—including red (OPT_R), green (OPT_G), and infrared (OPT_{IR})—and use the internal photodiode to measure light intensity on the surface of the skin, which can be affected by fluid volume in tissues while extravasation is occurring.

Regarding the power consumption of the sensing components, in this design the sensors are operate at very low duty-cycle to collect physical data, i.e., 1 Hz. For example, MAX30205 operates at 600μA in only a few milliseconds and sleep (3.5 μA) the remaining time. Therefore, the power consumption of sensing components is not dominated the lifetime of our design.

During the model training stage, the collected data are transferred by USB or Bluetooth to the data server for model training. Model inference can be executed on the data server or on a wearable device; the design of the model is already considered to inference model on a memory-constrained microprocessor-embedded system. This concern is discussed in the analysis of the validation experiment in the following subsection.

3.2 IV Simulation Platform

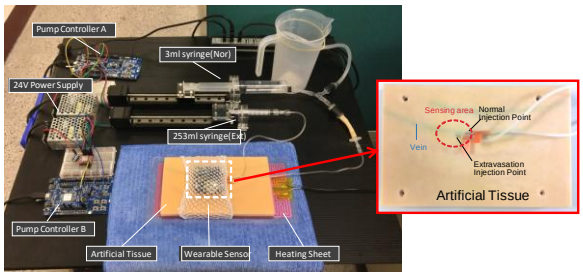

Fig. 5. The architecture of the simulation platform

In the development stage, we built a simulation platform to test and evaluate the performance of the wearable sensor and deep learning algorithm. We designed an IV experiment simulation platform, as shown in Fig. 5. An artificial skin tissue (or silicon-made injection training pad, which is for practicing venipuncture IV injection) was used to simulate human skin tissue and veins. A heating sheet was used to warm up the artificial skin tissue to 34 ℃ at the surface. The fluid temperature is close to the room temperature varying from 22 C to 26 ℃. This platform uses two IV pumps controlled by the different controllers to regulate the injection rate of two syringes, which inject the fluid into the vein for normal injection and outside the vein for extravasation injection. This design is intended to generate the normal and abnormal injections in a standard procedure in the artificial tissue without human intervention to swap the catheter tip and produce the symptoms in normal IV therapy and extravasation at the same location.

Fig. 6 shows an experiment to observe the swelling and temperature change of the skin with thermal imaging techniques by simulating the nonextravasation (normal) and extravasation IV therapy. In normal venipuncture IV therapy, the pump injects fluid into the vein and will not cause swelling skin (refer to Fig. 6a); skin temperature (refer to Fig. 6b) near the catheter tip does not also change.

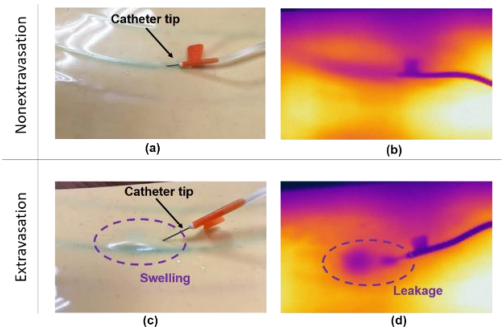

Fig. 6. The swelling and temperature change of the skin during nonextravasation (normal) and extravasation IV therapy. The darker color in the thermal images (b) and (d) indicates a lower temperature.

In an abnormal condition, the pump injects fluid into the tissue, as the catheter tip is outside the vein, and causes swelling near the catheter tip (refer to Fig. 6c); meanwhile, the leakage of fluid to the skin surface also causes skin temperature to drop significantly to lower than normal skin temperature (refer to Fig. 6d). This experiment's result confirms the observation in the previous study [6] and supports the use of BT in this design. The symptom of swelling can also be seen while extravasation occurs; hence, PS is integrated into the wearable sensor board to detect this symptom. Meanwhile, measuring the specific temperature level is less meaningful, as the measured temperature will be affected by the surrounding environment. What is more important is how the temperature changes over time. We used deep learning to identify the features of "temperature change" to detect extravasation but did not use a specific threshold, as other previous studies.

The proposed wearable sensor—which integrates BT, PS, and OPT (including OPT_R, OPT_G, and OPT_{IR})—is in contact with the skin near the catheter tip, with a fixation tape during IV therapy, and these sensors' data are collected at 1 Hz for later analysis.

As shown in Fig. 5, the platform has two syringes (the 25 ml syringe for normal IV injection and the 3 ml syringe for extravasation IV injection), which are separately controlled by IV pump controllers A and B. During the experiment, the injection rates of the two syringes are programmable from zero to 60 ml per h to simulate the IV therapy in normal conditions and extravasation.

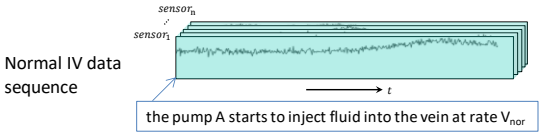

Fig. 7. The data sequence obtained during normal intravenous therapy

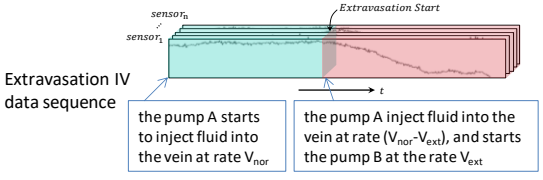

Fig. 8. The data sequence obtained during extravasation

Fig. 7 shows a normal IV data sequence from five sensors (BT, PS and OPT_R, OPT_G, and OPT_{IR}). In this case, we can regulate IV pump A to inject fluid into the vein at a predefined injection rate and duration. In the case we want to generate an

extravasation data sequence, IV pump A starts to inject fluid into the vein until extravasation starts; then we start IV pump B to inject fluid into the tissue outside the vein (the red part in Fig. 8). By controlling the timing to start/stop pumps A and B, we can generate several time-sequence data (with the values of five sensors) for the wearable sensor to simulate different scenarios of extravasation.

3.3. Data Server

The design goal of the data server is to determine the occurrence of extravasation; thus, the algorithm considers as a binary classification problem: extravasation/normal conditions. This design is based on supervised learning, as the time extravasation begins is controlled during the experiment. All the collected data are associated with a timestamp when extravasation begins. This proposed design model combines the feature of 1D-CNN (One-Dimensional Convolutional neural Network) and RNN (Recurrent Neural Network), using the convolution layer as a feature extractor to abstract the features of the input data from the multiple sensors. The recurrent layer of LSTM further adds the time-domain relationship information to the data; therefore, the output of the model correlates with the previous input data, not only the current input data. We call the proposed model lightweight convolutional LSTM (Light-ConvLSTM).

Two models were also proposed in this study for performance comparisons, a lightweight CNN (Light-CNN) model and logistic regression (LR). Light-CNN is similar to Light-ConvLSTM, both using a convolution layer as a feature extractor. The only difference is the output of Light-CNN is based on the current input only and all the outputs are independent, as it has no recurrent layer. LR is a simple statistical model that requires feature extractions from raw data to improve inference results; meanwhile, LR can be considered as a single-layer neural network, while Light-ConvLSTM is a multiple-layer neural network. The details are introduced in the following subsections.

3.3.1 Data Preprocessing

Data preprocessing is definitely required, as the raw data from sensors is often complicated and incomplete and in different units, with unclear features. The data preprocessing steps are outlined below:

1) **Time-series data segmentation:** The time-series raw data are segmented into fixed-length window data and feeds into the model during the training stage. The two parameters ‘window size’ and ‘window stride’ are used to adjust the data into the model, where window size controls the number of data points in each sequence and window stride controls the percentage of data overlapping between the previous and current sequences.
2) **Shuffle dataset by IV experiment index:** The segmented datasets are then randomly shuffled to prevent the dataset subset lacking representativeness of all the data and cause overfitting issues.
3) **Dataset splitting:** Splitting the dataset into a training set, validation set, and test set—which in this study are split into 80%, 10%, and 10%, respectively.
4) **Feature scaling:** The scaling of data from different sensors is done using equation (1):

$$z_{k,w}^{(i)} = \frac{x_{k,w}^{(i)} - \bar{x}_{k,w}}{\sigma} \quad (1)$$

where k is the index of the experiment, w is the index of the window in the dataset, and $\boldsymbol{x_{k,w}^{(i)}}$ indicates the data ith data point in the wth windowed data of the kth experiment. $\boldsymbol{\bar{x}_{k,w}}$ indicates the mean of all data in the same wth window, and $\boldsymbol{\sigma}$ is the mean of all standard deviation $\boldsymbol{\sigma_w}$ of windowed data from the normal (nonextravasation) data. The scaled data, $\boldsymbol{z_{k,w}^{(i)}}$, can help improve the rate of convergence.

3.3.2 The Light-ConvLSTM, Light-CNN, and LR Models

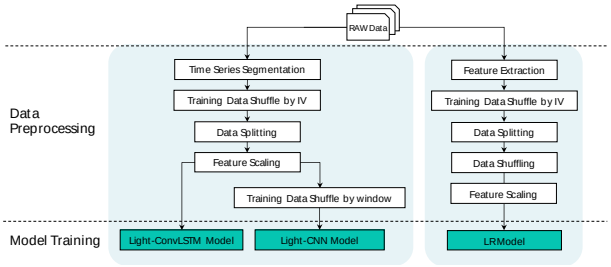

Fig. 9. The procedures of data preprocessing of the Light-ConvLSTM, Light-CNN, and logistic regression models.

Fig. 9 shows the procedures of the raw data preprocessing for three analytical models in this experiment: the Light-ConvLSTM, Light-CNN and LR models. The performance of the proposed Light-ConvLSTM was compared with the performances of the latter two models.

3.3.2.1 Light-ConvLSTM

As shown in Fig. 10, Light-ConvLSTM is composed of three convolution layers, one average pooling layer, two LSTM recurrent layers, two fully connected layers, and one softmax layer. In the convolution section, the raw data input from multiple sensors (in this design, the five sensors) are converted into abstracted feature data, and the activation function is ReLU. Average pooling is adapted in the pooling layer to preserve more details of the raw data changes. The flatten section is composed of one flatten layer and one fully connected layer. As the 2D feature map from the convolution layer is converted to a 1D flatten layer, the time-sequence relationship is lost. Here, we used a fully connected layer to increase the model complexity. The recurrent section is composed of two LSTM layers, which indicates the next output of the LSTM correlates with the current and past features from the previous LSTM memorized in the cells. In this section, we used two LSTM layers, which was demonstrated in Karpathy et al.’s[14] study that this can improve performance better than using a single LSTM layer. The decision section is composed of one fully connected layer and one softmax layer (which was designed according to Sainath et al.’s [9] study). This section converts the features into two categories: the posterior probability of

extravasation (denoted by P_{ext}) and normal IV (denoted by P_{nor}).

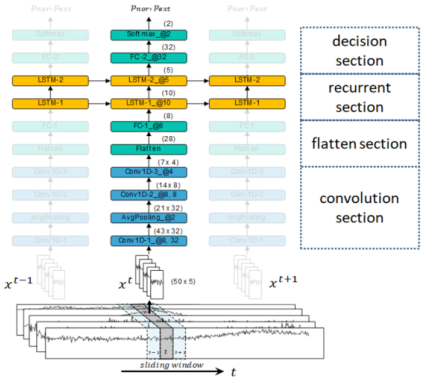

Fig. 10. The structure of the Light-ConvLSTM

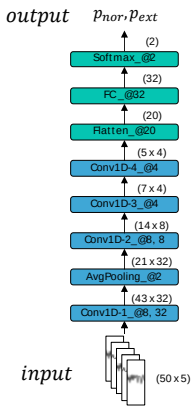

Fig.11. The structure of the Light-CNN

3.3.2.2 Light-CNN

Shown in Fig.11, Light-CNN is simplified Light-ConvLSTM without recurrent layers; therefore, Light-CNN does not have memory cells as Light-ConvLSTM does. Data preprocessing for Light-CNN is the same as that of Light-ConvLSTM, except an additional data shuffle is applied to avoid sequential data causing a biased estimation.

3.3.2.3 Logistic Regression

LR is a simple statistical regression model and can be considered as a single-layer single-unit neuronal network. LR data preprocessing is similar to Light-CNN's, except each input of LR to represent the data from five sensors is a 5 × 1 vector.

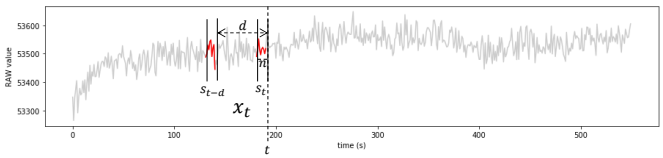

Fig. 12. The feature extraction of logistic regression

$$s_r^k(t) = \frac{1}{n}\sum_{i=0}^{n-1} p_r^k(t-i) \quad (2)$$

$$x_r^k(t) = \frac{s_r^k(t) - s_r^k(t-d)}{s_r^k(t-d)} \quad (3)$$

As the right portion of Fig. 9 shows the raw data preprocessing of LR, the first step is feature extraction. Fig. 12 shows the feature extraction of LR, where n is the number of the data (window size) to retrieve from the raw data, d is the defined step-size, s is the 'sensor response', and x is the sensing feature. Equation (2) describes the calculation of sensor response $s_r^k(t)$, where k is the number of IV experiment, r is the index of five sensors (from 1 to 5, corresponding with PS, BT, OPT_R, OPT_{IR}, and OPT_G), and $p_r^k(t)$ indicates the value of sensor r at time t in the IV experiment k.

As shown in equation (3), the sensor response is then used to calculate the sensor feature $x_r^k(t)$, which indicates the difference between sensor response $s_r^k(t)$ and $s_r^k(t-d)$ of sensor r at time t in the IV experiment k. All the sensor features are inputs of the model, as shown in Fig. 13. The output of the model, $p(c|x)$, is the posterior probability of occurring extravasation.

$$f_{w,b}(x) = \sigma(b + \sum_r w_r x_r) = p(c \mid x) \quad (4)$$

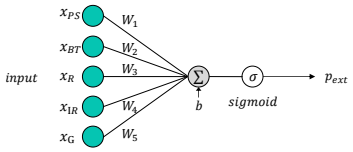

Fig. 13. The representation of the logistic regression model

IV. Experiment and Analysis

4.1 Experimental Design

To evaluate the proposed design, before clinical experiments, we designed an experiment to collect the data from the wearable sensor and evaluate the proposed model with the proposed IV simulation platform, as shown in Fig. 5. This platform can simulate the physical change of human skin during IV therapy by injecting the fluid (water) into the artificial skin. Then the wearable sensor collects the data sequences during the IV therapy. In this experiment, there were two simulating cases: under **normal IV** conditions and during **extravasation**.

1) For the **normal IV case**, referring to the data sequence example shown in Fig. 7, pump A, in Fig. 5, injected the fluid into the vein in the artificial tissue at rate V_{nor} during the entire IV therapy session. In these experiments, V_{nor} was in the range of 10 mL/h to 60 mL/h, the period of the experiment was from 5 to 30 min.

2) For the **extravasation case,** referring to the data sequence example shown in Fig. 8, pump A, in Fig. 5, started to inject fluid into the vein at rate V_{nor} for a period; when extravasation begins, pump A's rate was lowered to V_{nor}-V_{ext} and started pump B at rate V_{ext} (in the range of 1 mL/h to 15 mL/h). This procedure was intended to

simulate a single syringe injecting the fluid into the artificial skin at a constant injection rate and generate the effects of extravasation without the need to manually relocate the catheter tip, making the whole process automatic.

Fig. 14. The artificial skin installed on the arm of participants during the experiments

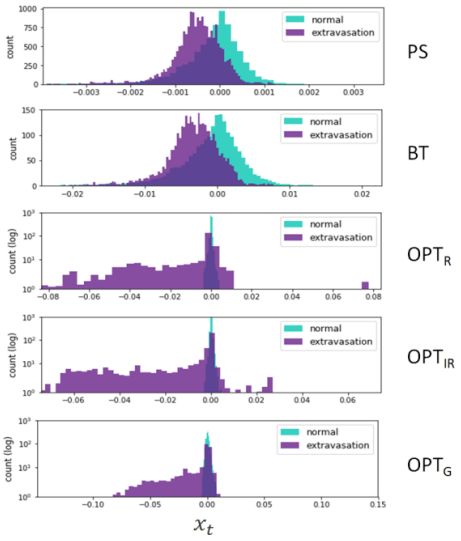

Fig. 15. The comparison of the data distributions from normal IV and extravasation data sequences

As in the stand-alone test, if the artificial skin and platform were installed on a table, the data collected from the wearable sensor will be too stable. Therefore, we installed the artificial skin on their lower arms to simulate the condition of the possibility of the arm moving unintentionally and causing noises that may affect the sensor reading in practical usage, and gender, arm circumference, and so on should not affect the experiment results. We installed the artificial skin and the wearable sensor on the arm of 30 subjects to collect data that are closer to what is obtainable in real situations. There are 22 types of IV parameter settings, which are listed in Table 1. Executing each parameter setting on a subject generates an IV data sequence. Subjects were randomly assigned to these settings. We collected 238 IV data sequences (including 165 for extravasation and 73 for normal IV conditions) on this simulation platform.

The pictures in Fig. 14 show the participants with the artificial skin during the experiments to simulate real IV therapy under the conditions of normal and extravasation. Each IV therapy generated a data sequence containing five sensors' readings: PS, BT, OPT_R, OPT_{IR}, and OPT_G. The total time period for all the data sequences was 50.7 h, including normal IV therapy for 37.6 h and extravasation for 13.1 h.

Fig. 15 shows the comparison of data distributions from a normal IV therapy case and an extravasation case. This figure shows the differences between these two data distributions were significant; practically, the distributions of optical sensors (red, IR and green) were significantly different, implying the optical sensors can be useful for detecting extravasation.

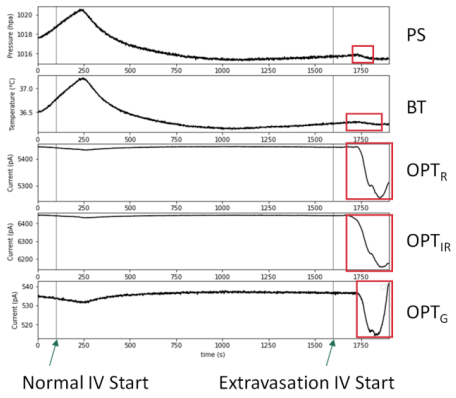

Fig. 16. The representation of the five sensor channels of an extravasation IV data sequence

Fig. 16 shows an IV data sequence example for extravasation. At time 100 s, pump A started to inject fluid into the vein to simulate normal IV injection, and then extravasation occurs at time 1600 s by starting pump B. We can observe that the heating sheet below the artificial skin caused the temperature to rise at the beginning, and the temperature drops at 250 s as the fluid in the vein starts to cool down the artificial skin. However, PS only detects the pressure changes slightly caused by the swelling. In addition, the reading of BT changes was not significant. The readings of three optical sensors (OPT_R, OPT_{IR}, and OPT_G) all dropped down significantly after extravasation occurred, marked by the red rectangles in Fig. 16. This might be because the skin swelling affects the optical sensors could not firmly contact the artificial skin. In summary, this example demonstrates the optical sensors might be more useful in detecting extravasation than the temperature and pressure sensors, but they can still be used to compensate for the other sensors to increase the detection rate and lower the false alarm rate.

4.2 Analysis

This section analyzes the collected data based on the LR, Light-CNN, and the proposed Light-ConvLSTM models. We used Keras [15] to implement Light-CNN and Light-ConvLSTM and used scikit-learn [16] to implement LR. Table 2–Table 5 show all the hyperparameters and parameters (including dataset splitting) used in the experiments.

4.2.1 Analysis of the Performance of Logistic Regression

Here, we analyze the results of the experiment. The performance of the algorithm was evaluated by comparing the prediction and the ground truth. Table 6 shows the test results of LR—*d* is a feature of LR, illustrated in Fig. 12. Referring to the case $d = 50$ for the 'single sensor group', which only used

the data from a single sensor to detect the event, the detection rate, P_d, of PS and BT were both above 60% but accompanied with a false alarm rate, P_f, above 30%. For optical sensors, the P_f of OPT_R and OPT_{IR} were all less than 0.5%, but their P_d were also less than 40%. 'OPT_G only' performed the worst among all the cases, as its F1 score was the lowest in this evaluation (only 0.268), which might be because the green spectrum has a lower penetration depth into the tissue.

Regarding the 'multisensor group', for the combination of using PS and BT, the results were similar to using a single sensor, PS or BT. This might be because the characteristics of these two sensors are very similar, and hence, there are no gained benefits from fusing their data. For the case combining all the OPT sensors (denoted by 'All OPT'—including OPT_R, OPT_{IR}, OPT_G), the F1 score was 10% higher when compared to the single OPT sensor; this is because the P_d of 'All OPT' increased to 48.2%—compared to the P_d of OPT_{IR}, which was only 36.7%. This shows these optical sensors are capable of compensating one other to increase the detection rate. As the F1 score of 'OPT_G only' was only 0.268, we further investigated how including or excluding OPT_G would affect and show the results in Table 7. The results show OPT_G almost has no effect on the F1 score. This demonstrates LR is robust to deal with the data from some ineffective sensors. Meanwhile, referring to Table 6 , the F1 score of the 'All sensors' group was 0.588, which was lower than the F1 score the 'All OPT' group (0.638). Meanwhile, the P_f of the 'All sensors' group was 11.7%, which was 10% higher than that of the 'All OPT' group—even though the P_d of the 'All sensors' group was 57.3%, which was higher than the P_d of the 'All OPT' group. This points out PS and BT contribute some data, helping to increase the detection rate and lower the false alarm rate.

Comparing d = 50 and d = 100 in Table 6 , which indicates the time interval of sample two data (refer to Fig. 12), the larger d indicates a longer time interval and should result in a more considerable difference if extravasation occurs in an IV data sequence. However, the result shows the improvement of a larger d was not significant. In summary, LR is not a good model to detect extravasation; nevertheless, the multisensor group performed significantly better than the single sensor group, even with data from some an ineffective sensor (i.e., OPT_G).

4.2.2 Analysis of the Performances of the Light-CNN and Light-ConvLSTM Models

In the following, we discuss the results of the analysis of the performance of the Light-CNN and Light-ConvLSTM models. Table 8 shows the analysis of the performance of the Light-CNN and Light-ConvLSTM, which are obtained from the same configuration.

The results of the **Light-CNN** model was similar to the LR model: the 'OPT_G only' was the worst among all the other cases in single sensor group and P_d of PS and BT were also higher—70.9% and 74.1%, respectively—but accompanied by higher P_f, 24.9% and 38.6%.

For the proposed **Light-ConvLSTM** model, the P_d of PS and BT further improved to 82.8% and 74.4% and lower their P_f, down to 18% and 18.7%. This is because LSTM memory cells can help retain the previous information and are more suitable to detect the event of interest in long-term sequential data. Similar to LR, for both Light-CNN and Light-ConvLSTM, the 'multisensor group' consistently outperformed the 'single sensor group'. As expected, the highest P_d and the lowest P_f were observed by 'all sensors' in the Light-ConvLSTM model—83.7% and 6.2%, respectively—and resulted in the highest F1 score (0.828). This result supports the proposed Light-ConvLSTM model can correctly detect extravasation, with minimal false alarm probability.

It is worthy to mention that the F1 score of 'OPT_{IR} only' (which is similar to ivWatch [18] with infrared sensor) in the Light-ConvLSTM model was 0.802, which is comparable to the 'All sensors' but with a little bit higher P_f (10.8%). This hints us that an 'OPT_{IR} only' wearable device might be good enough and useful to detect extravasation if the hardware cost is particularly limited. However, if a lower P_f is required, our design with multiple sensors should be a better solution.

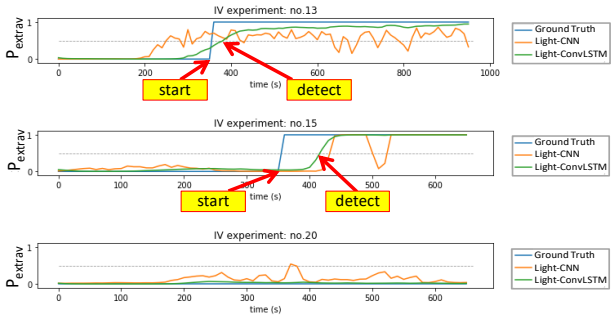

Fig. 17. The comparison of extravasation detection by Light-CNN and Light-ConvLSTM

Fig. 17 shows three IV data sequence examples to demonstrate the performance of Light-CNN and Light-ConvLSTM. The vertical axis, P_{extrav}, indicates the probability of extravasation occurring. The top two examples are IV data sequences with extravasation and the bottom one is a normal IV data sequence. We can see the Light-CNN could detect extravasation was occurring by referring to the ground truth blue line, but the output probability was fluctuating and less stable. The green lines output by the proposed Light-ConvLSTM model were a closer fit to the ground truth and were more smooth and stable than Light-CNN's. This again confirms the proposed Light-ConvLSTM is suitable and can be used to reliably detect extravasation.

With the Light-ConvLSTM algorithm and set the threshold of P_{extrav} to 50%, we can see in experiment no. 13, that the extravasation began at about 350 s (the starting time corresponding to syringe B) and was detected at about 390 s ($P_{extrav;}$ the probability of extravasation is bigger than 50%); in experiment no. 15, the extravasation began at about 350 s and was detected at about 410 s. In these two experiments, the latency from the time the extravasation started (note that "start" indicates begin to inject fluid from syringe B, not the time the extravasation is observable with naked eyes) to the time the event was detected was about 40–60 s. Based on this, we defined the threshold of P_{extrav} as 50%. If we lower the threshold, we can reduce the latency but might also incur more false alarms. To set a reasonable threshold of P_{extrav}, we need to collaborate with professional nursing staff regarding clinical

experiments, and can be adjusted by the users to strike the balance between latency and false alarm.

4.2.3 Analysis of the Light-ConvLSTM Model with Different Hyperparameters

Here, we further analyze how the hyperparameters of the Light-ConvLSTM model may affect its performance. The 'window size' affects how many data samples extracted from a data sequence at one time can train the model and predict the outcome. Table 9 depicts Light-ConvLSTM performs the best while the window size is 25, while its P_d is 85.5% and P_f is 4.1%. Using a larger window size, such as 100, would lower the P_d to 79.2% and raise the P_f to 7.7%. This is because a larger window size will include more data samples in a window and dilute the data change caused by the effects of extravasation. Using a smaller window size also brings the benefit that the model consumes less memory and computation complexity; therefore, we set the window size to 25 in the following analysis.

Table 10 illustrates the performance of the Light-ConvLSTM model using a different 'sampling rate' for the sensors. For an embedded system such as the wearable device in this study, the lower sampling rate to read data from the sensors indicates lower power consumption and less memory usage. This table shows that even when the sampling rate was down to 0.125 Hz or 1 sample per 8 s, F1 scores were all similar and their P_d and P_f did not fluctuate significantly. Therefore, if the power and memory consumption is extremely limited, setting the sample rate to a lower rate, such as 0.125 Hz, is acceptable while maintaining good performance.

We know the larger 'window stride' results in less overlap between the data of two consecutive windows. Table 11 shows different window strides did not affect the detection performance very significantly, and F1 scores were still somehow similar. Therefore, setting the window stride to 100 is acceptable if the design is intended to minimize resource usage.

Table 12 evaluates how different 'polling modes' affect performance. The results show average pooling can improve all results over max pooling, as average pooling can preserve more details of the raw data changes.

Table 13 changes the number of 'convolution layers' of the Light-ConvLSTM model to evaluate the detection performance changes. The result shows the number of convolution layers does not affect the P_d and P_f significantly; however, using more convolution layers will reduce the size of the feature map connected to the flatten layer as well as reduce the parameter size. For example, the parameter size of default structure of Light-ConvLSTM (composed of three convolution layers, two LSTM recurrent layers) is 5228; the flash and RAM required to store the parameter arrays are 5228 x 4 byte (the size of float is 4 bytes) = 20912 bytes or 20.4 kB, which is small and can be stored in the memory of the wearable sensor. Therefore, if the embedded system's memory is small, we can use more convolution layers to lower the memory required to store the model without scarifying the performance. Meanwhile, we also tested the run time of the proposed model by run the model inference for 50 times to obtain the averages. A desktop computer with Intel i7-4712MQ CPU needs 1.5 ms to run a single inference, as well as the proposed wearable device requires 39.2 ms, which is still a very short period and will not be noticeable by users. This result again support the proposed model is suitable to run not only on a desktop computer but also in a resource-constrained embedded system.

V. Discussion and Conclusion

This study presents an unprecedented extravasation detection system—including the wearable sensor, the simulation platform, and the data server for analyzing data to output the probability of occurring extravasation. We propose a Light-ConvLSTM model to detect extravasation with a high detection rate and low false alarm rate. We conducted a comprehensive evaluation to show the feasibility and benefits of the proposed design. By using the data from all the sensors on the sensor board and Light-ConvLSTM model, the detection rate can be as high as 83.7% while keeping the false alarm rate at only 6.2%. In conclusion, our design is open-source, including the hardware design and software algorithms, so those interested in this work can further reproduce the experiment in vivo to improve our design. As far as we know, this is the first open-source work considering both integrating multiple sensors and deep learning algorithms to detect extravasation with systematic evaluation. We hope this design can be further improved and eventually used in hospitals to help detect extravasation early and reduce harm to patients.

Acknowledgements

The authors would like to thank Sing-Long Lin from Tzi-Chi Hospital Da-Lin Branch for their valuable comments. The authors would like to thank Pei-Jyi Lee and Pin-Chen Kuo for their excellent assistance.

References

[1] M.-Y. Cheng et al., "Wearable Sensor Patch for Early Extravasation Detection," in 2016 IEEE 66th Electronic Components and Technology Conference (ECTC), 2016: IEEE, pp. 1632-1637.

[2] Y. Matsui et al., "Evaluation of the Predictive Validity of Thermography in Identifying Extravasation With Intravenous Chemotherapy Infusions," Journal of Infusion Nursing, vol. 40, no. 6, p. 367, 2017.

[3] Y.-C. Du, W.-T. Chen, W.-S. Ciou, and C.-M. Tsai, "A novel device for non-invasive assessment of extravasation during injection by NIRS technology," in 2017 IEEE SENSORS: IEEE, pp. 1-3.

[4] A. O. Bicen, L. L. West, L. Cesar, and O. T. Inan, "Toward non-invasive and automatic intravenous infiltration detection: evaluation of bioimpedance and skin strain in a pig model," IEEE journal of translational engineering in health and medicine, vol. 6, pp. 1-7, 2018.

[5] M.-Y. Cheng et al., "Wearable Sensor Patch for Early Extravasation Detection," in 2016 IEEE 66th Electronic Components and Technology Conference (ECTC), 2016: IEEE, pp. 1632-1637.

[6] Y. Matsui et al., "Evaluation of the Predictive Validity of Thermography in Identifying Extravasation With Intravenous Chemotherapy Infusions," Journal of Infusion Nursing, vol. 40, no. 6, p. 367, 2017.

[7] Y.-C. Du, W.-T. Chen, W.-S. Ciou, and C.-M. Tsai, "A novel device for non-invasive assessment of extravasation during injection by NIRS technology," in 2017 IEEE SENSORS: IEEE, pp. 1-3.

[8] A. O. Bicen, L. L. West, L. Cesar, and O. T. Inan, "Toward non-invasive and automatic intravenous infiltration detection: evaluation of bioimpedance and skin strain in a pig model," IEEE journal of translational engineering in health and medicine, vol. 6, pp. 1-7, 2018.

[9] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 4580-4584.

[10] "nRF52840 - Nordic Semiconductor," nRF52840 - High-end Bluetooth low energy System-on-chip. Online]. Available:

https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF52840. [Accessed: 29-Jan-2020].

[11] "LPS22HB," STMicroelectronics. [Online]. Available: https://www.st.com/en/mems-and-sensors/lps22hb.html. [Accessed: 29-Jan-2020].

[12] "MAX30205 Human Body Temperature Sensor," Maxim. [Online]. Available: https://www.maximintegrated.com/en/products/sensors/MAX30205.html. [Accessed: 29-Jan-2020].

[13] "MAX30105 High-Sensitivity Optical Sensor for Smoke Detection Applications," Maxim. [Online]. Available: https://www.maximintegrated.com/en/products/interface/sensor-interface/MAX30105.html. [Accessed: 29-Jan-2020].

[14] A. Karpathy, J. Johnson, and L. Fei-Fei, "Visualizing and understanding recurrent networks," arXiv preprint arXiv:1506.02078, 2015.

[15] "Keras: The Python Deep Learning library," Home - Keras Documentation. [Online]. Available: https://keras.io/. [Accessed: 02-Feb-2020].

[16] "learn: machine learning in Python - scikit-learn 0.16.1 documentation," *scikit*. [Online]. Available: https://scikit-learn.org/. [Accessed: 02-Feb-2020].

[17] Jheng-Sing Lin, Che-Wei Kuo, Wei-Chen Huang, Huang-Chen Lee, A Sensor-Fusion Approach System for Detecting Early Extravasation of Infant Intravenous Infusion, The 18th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN 2019), Demo Session, Montreal, Canada, April 2019.

[18] "ivWatch Products: ivWatch Model 400: Continuous IV Monitoring," ivWatch. [Online]. Available: https://www.ivwatch.com/products/. [Accessed: 13-Apr-2020].

[19] van Rens, M., Hugill, K., Francia, A.V., "A new approach for early recognition of peripheral intravenous (PIV) infiltration: A pilot appraisal of a sensor technology in a neonatal population", Vascular Access, vol. 5,issue 2, pp. 38-41, Oct 2019.

Table 2 LR hyperparameters

	Hyper-parameter	Setting
Pre-processing	n	10
	d	50
	Dataset (train : test)	0.8 : 0.2
Modeling	Loss Function	Cross-entropy
	Update Rule	L-BFGS

Table 3 Light-CNN, Light-ConvLSTM hyperparameters

	Hyper-parameter	Setting
Pre-processing	Window Size	50
	Window Stride	10
	Dataset (train : valid : test)	0.8 : 0.1 : 0.1
Modeling	Loss Function	Cross-entropy
	Update Rule	Adam (with default setting*)
	Batch size (Light-ConvLSTM)	1
	Batch size (Light-CNN)	32
	Epoch	25

** learning_rate=0.001, beta_1=0.9, beta_2=0.999*

Table 4 Light-ConvLSTM parameters

Layer	**Parameter**	**#Para.**
Conv1D-1	f: $8 * 5 * 32$ b: 32	1,312
AvgPooling	0	0
Conv1D-2	f: $8 * 32 * 8$ b: 8	2,056
Conv1D-3	f: $8 * 8 * 4$ b: 4	260
Flatten	0	0
FC-1	W: $28 * 8$ b: 8	232
LSTM-1	W_x: $8 * 10 * 4$ W_h: $10 * 10 * 4$ b_x: $10 * 4$ c: 10 h: 10	780
LSTM-2	W_x: $10 * 5 * 4$ W_h: $5 * 5 * 4$ b_x: $5 * 4$ c: 5 h: 5	330
FC-2	W: $5 * 32$ b: 32	192
Softmax	W: $32 * 2$ b: 2	66
Total		5228

Table 5 Light-CNN parameters

Layer	**Parameter**	**#Para.**
Conv1D-1	f: $8 * 5 * 32$ b: 32	1,312
AvgPooling	0	0
Conv1D-2	f: $8 * 32 * 8$ b: 8	2,056
Conv1D-3	f: $8 * 8 * 4$ b: 4	260
Conv1D-4	f: $3 * 4 * 4$ b: 4	52
Flatten	0	0
FC	W: $20 * 32$ b: 32	672
Softmax	W: $32 * 2$ b: 2	66
Total		4418

Table 6 Analysis of LR

		d=50				d=100			
		Acc	P_d	P_f	F1 score	Acc	P_d	P_f	F1 score
Single sensor group	PS only	67.7%	63.1%	30.9%	0.481	68.8%	64.6%	29.7%	0.514
	BT only	67.2%	64.2%	31.8%	0.486	68.3%	64.0%	30.2%	0.508
	OPT_R only	83.9%	32.9%	0.2%	0.492	82.5%	32.2%	0.2%	0.485
	OPT_{IR} only	84.5%	36.7%	0.4%	0.532	83.7%	37.7%	0.8%	0.538
	OPT_G only	69.4%	23.6%	16.4%	0.268	58.2%	41.3%	36.0%	0.338
Multi sensor group	PS+BT	67.3%	63.5%	31.5%	0.481	68.7%	63.9%	29.6%	0.509
	All OPT	87.1%	48.2%	0.9%	0.638	85.8%	48.6%	1.4%	0.637
	All sensors	80.9%	57.3%	11.7%	0.588	80.3%	57.7%	12.1%	0.595

Table 7 Analysis of LR with or without OPT_G

	d=50			
	Acc	P_d	P_f	F1 score
All OPT	87.1%	48.2%	0.9%	0.638
OPT_R + OPT_{IR}	87.1%	48.2%	0.7%	0.641
All	80.9%	57.3%	11.7%	0.588
All without OPT_G	81.0%	57.2%	11.6%	0.588

Table 8 Analysis of Light-CNN and Light-ConvLSTM

		Light-CNN				Light-ConvLSTM			
		Acc	P_d	P_f	F1 score	Acc	P_d	P_f	F1 score
Single sensor group	PS only	73.5%	70.9%	24.9%	0.504	81.6%	82.8%	18.0%	0.641
	BT only	64.0%	74.1%	38.6%	0.453	80.3%	74.7%	18.7%	0.601
	OPT_R only	79.4%	55.2%	11.0%	0.593	85.2%	79.4%	12.6%	0.747
	OPT_{IR} only	79.3%	67.8%	16.4%	0.649	88.3%	84.6%	10.8%	0.802
	OPT_G only	80.9%	29.9%	1.7%	0.442	80.6%	68.1%	14.6%	0.639
Multi sensor group	PS+BT	57.1%	77.3%	49.8%	0.478	78.4%	39.7%	8.9%	0.427
	All OPT	82.0%	76.6%	17.1%	0.688	90.6%	81.0%	7.0%	0.814
	All sensors	84.3%	83.8%	15.6%	0.727	91.3%	83.7%	6.2%	0.828

Table 9 Analysis of Light-ConvLSTM with different windows size

		Light-ConvLSTM			
		Acc	P_d	P_f	F1 score
Window size	25	93.3%	85.5%	4.1%	0.860
	50	91.3%	83.7%	6.2%	0.828
	100	88.8%	79.2%	7.7%	0.782

Table 10 Analysis of Light-ConvLSTM with different sample rate

		Light-ConvLSTM			
		Acc	P_d	P_f	F1 score
Sample rate (HZ)	1	91.3%	83.7%	6.2%	0.828
	0.5	88.9%	72.3%	3.5%	0.826
	0.25	91.5%	82.8%	5.0%	0.843
	0.125	92.5%	80.2%	3.5%	0.839

Table 11 Analysis of Light-ConvLSTM with different window stride

		Light-ConvLSTM			
		Acc	P_d	P_f	F1 score
Window stride	10	91.3%	83.7%	6.2%	0.828
	20	88.6%	77.6%	7.4%	0.780
	50	94.6%	83.4%	2.3%	0.869
	100	92.8%	83.3%	4.4%	0.839

Table 12 Analysis of Light-ConvLSTM with different pooling mode

		Light-ConvLSTM			
		Acc	P_d	P_f	F1 score
Pooling mode	Avg.	92.2%	80.9%	4.7%	0.823
	Max.	90.5%	79.1%	6.3%	0.794

Table 13 Analysis of Light-ConvLSTM with different layers configurations

	Light-ConvLSTM				
Layers (CNN, STM)	Acc	P_d	P_f	F1 score	Parameter size
(1, 2)	92.3%	81.9%	4.2%	0.843	13696
(2, 2)	93.4%	82.2%	2.8%	0.866	5896
(3, 2)	91.3%	83.7%	6.2%	0.828	5228