Blood leakage detection during dialysis therapy based on fog computing with array photocell sensors and heteroassociative memory model

Jian-Xing Wu[1], *Ping-Tzan Huang*[2], *Chia-Hung Lin*[3,4] ✉, *Chien-Ming Li*[5]

[1]*Niche Biomedical LLC, California NanoSystems Institute at UCLA, Los Angeles 90095, CA, USA*
[2]*Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan*
[3]*Department of Electrical Engineering, Kao-Yuan University, Kaohsiung City, 82151, Taiwan*
[4]*Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung City, 41170, Taiwan*
[5]*Division of Infectious Diseases, Department of Medicine, Chi Mei Medical Center, Tainan City 710, Taiwan*
✉ *E-mail: eechl53@gmail.com*

Published in Healthcare Technology Letters; Received on 4th October 2017; Revised on 23rd November 2017; Accepted on 8th December 2017

Blood leakage and blood loss are serious life-threatening complications occurring during dialysis therapy. These events have been of concerns to both healthcare givers and patients. More than 40% of adult blood volume can be lost in just a few minutes, resulting in morbidities and mortality. The authors intend to propose the design of a warning tool for the detection of blood leakage/blood loss during dialysis therapy based on fog computing with an array of photocell sensors and heteroassociative memory (HAM) model. Photocell sensors are arranged in an array on a flexible substrate to detect blood leakage via the resistance changes with illumination in the visible spectrum of 500–700 nm. The HAM model is implemented to design a virtual alarm unit using electricity changes in an embedded system. The proposed warning tool can indicate the risk level in both end-sensing units and remote monitor devices via a wireless network and fog/cloud computing. The animal experimental results (pig blood) will demonstrate the feasibility.

1. Introduction: According to the 2016's statistics in Taiwan, more than 80,000 patients with end-stage renal disease and chronic kidney failure have being regularly received haemodialysis treatment. Venous needle dislodgement (VND) and blood leakage/blood loss are frequently observed clinically serious complications occurring during dialysis therapies. According to the American Nephrology Nurses' Association VND survey reports, more than 50% of patients on dialysis were concerned about VND or serious blood loss, indicating that they were concerned about VND events very often (>30%), often (>20%), or occasionally during dialysis therapy. More than 75% of the surveyed patients indicated that they had observed a VND event, and more than 8% had observed five events or more in the last 5 years [1]. These events are life-threatening complications. At a dialysis flow rate of 400–500 ml/min, an adult can lose more than 40% of blood volume in just a few minutes. The critical risk level is defined as the reaction time of <2.5 min at a blood flow rate of >200 ml/min. Therefore, an additional assistant tool is required for the detection of early blood leakage during dialysis therapies, as shown by the fog computing framework in Fig. 1*a*.

Currently, blood leakage and bleeding detection sensors such as pad sensors, wetness sensors, and optical sensors are being used to design early warning detectors for dialysis therapy, intravenous therapy, and wound healing. Customised products such as Redsense® monitor (Conformite Europeene Mark, Halmstad, Sweden, approved by the Food and Drug Administration) [2–4] and HEMOdialert™ (Anzacare) [5] have been applied for routine dialysis therapy, high-risk patients, and home dialysis treatment. Redsense® monitor is based on an optical fibre that transmits light to an optical detector. When blood leakage covers the sensing unit, the infrared (IR) light will be interrupted to identify the electrical changes. However, the IR light source is easily affected by moisture/sweat and temperature, and the intensity of the scattered IR light will be reduced. In addition, the IR light sources can be arranged as a single sensor or as an array of sensors. These optical sensors are continually monitored during dialysis therapy (3–4 h), while requiring continuous electrical power supply.

The HEMOsensor™ and HEMOdialert™ systems consist of two-spaced components. The sensing and alarm units are designed with analogue circuits in two-spaced components and require a line connecting an alarm, as shown in Fig. 1*b*. This system can be used to offer home dialysis treatment to patients. The wetness sensor detects direct current electricity changes (short circuit) when the blood contacts the electrodes. Leakage/loss of blood or other conductive liquids can be detected. However, additional analogue circuits and electronic devices may limit the patient's movements in a dialysis setting and also cause the patient to be restless, stressed, and worried about moving. Other wetness and pad sensors are sensitive to saline, blood, or conductive liquids [6, 7]. Although these sensors are simple to use and are of low cost with a unique electrical circuit, they have no warning indications regarding the Bluetooth/WiFi wireless communication and cannot stop the blood roller pump.

Therefore, based on fog computing (edge computing) with an array sensor, an assistant warning tool is used to design a connecting network, including one or more end-sensing units and a remote monitor system in an indoor haemodialysis room. A flexible sensor comprises an array sensor with four photocell sensors and a microdistribution connection circuit (metallic material) on the plastic substrate. The analogue circuit is fabricated via a screen-printing technique by printing the electronic circuits and mounting the circuit elements. Hence, the analogue circuitry of the sensing unit can be reduced. Its substrate is employed to detect the voltage changes with varying light intensities. The fog computing (edge computing) is a technique of optimising cloud computing by performing data processing at the edge of the wire/wireless communication network. The proposed framework can primarily analyse the time-sensitive data at the network edge or near the source of the data, instead of sending vast amount of data to the cloud [8]. While a sensing unit detects abnormal data for edge computing, the wireless transmitter sends the selected messages to the cloud for further analysis and storage. This technique can reduce the communication bandwidth between the sensor and the central data centre. In addition, this design may not be continuously

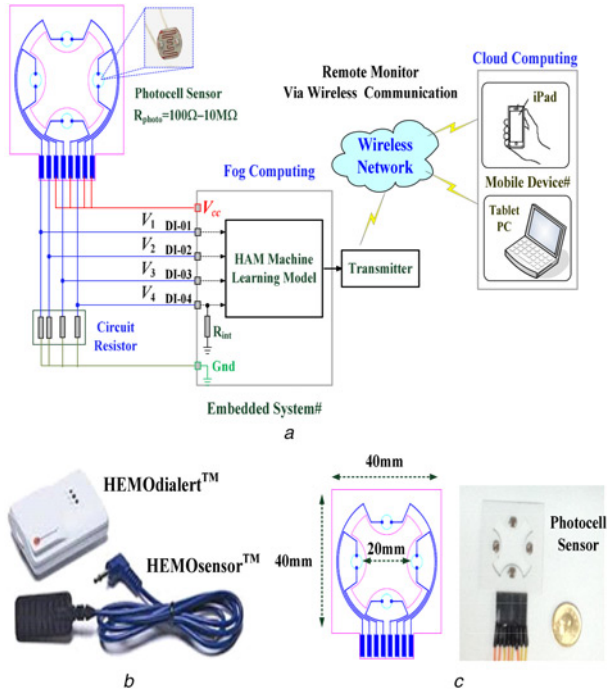

Fig. 1 *Proposed assistant tool for blood leakage detection*
a Fog computing framework
b Customised product
c Array photocell sensor based on fog computing

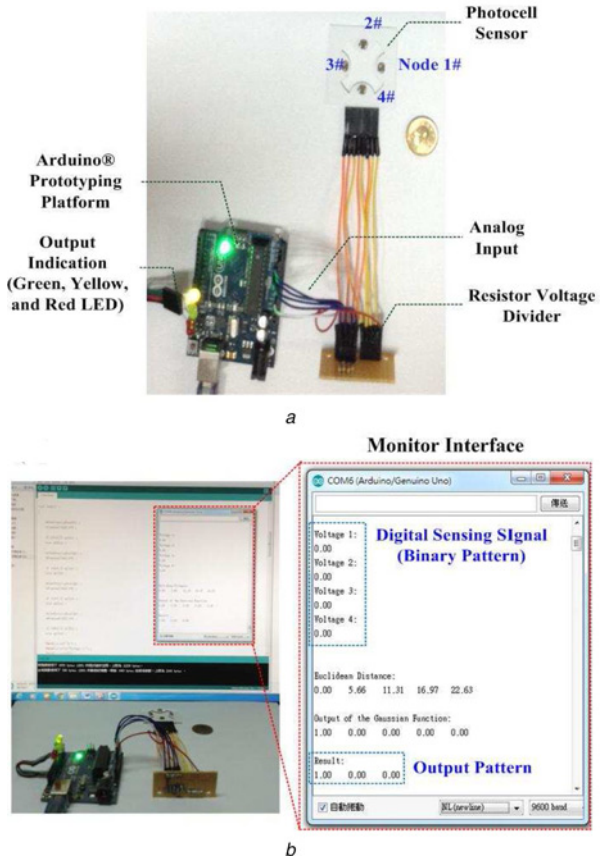

Fig. 2 *Fog computing based prototyping platform*
a Proposed fog computing by photocell sensors and Arduino prototyping platform
b Monitor interface

connected to a network and covers wireless sensor network, mobile data acquisition, and mobile signature analysis [9, 10]. Therefore, this framework can continuously monitor the real-time personalised health condition and can also integrate with the wireless sensor and the intelligent mobile device for use in the haemodialysis room.

In this study, four photocell sensors are arranged in an array configuration on a plastic substrate, as shown in Fig. 1*c*. A photocell sensor is a variable resistance semiconductor. It is a light-dependent resistor with varying light intensities. The resistor voltage divider and the voltage follower can be used to transfer voltage changes in the sensing unit. The sensing unit with the four photocell sensors has a multiposition switch function that manipulates high and low voltage levels. Then, a digital analytical design is employed to identify the leakage levels of blood or conductive liquids using hard limit function. In addition, the plastic substrate is thin and soft, and can be placed on the puncturing site (arterial or venous site) and be covered with the whole swab. In the alarm unit, a heteroassociative memory (HAM) model [11, 12] is employed to design a digitised alarm to automatically identify blood leakage levels. This machine learning model can deal with input binary pattern and acts as the biological behaviour to associate the possible risk level, the corresponding output binary pattern with the logic high signal to directly drive an alarm unit, such as triggering a loud alarm or a light-emitting diode (LED). The proposed HAM intelligent algorithm for fog computing can improve the detection reliability and can be easily implemented using a high-level programming language in an embedded system or a mobile device, such as Arduino® (Uno) prototyping platform, as shown in Fig. 2*a*. Hence, the analogue analytic circuitry of the alarm unit can be reduced. An end-sensing unit can become more intelligent to indicate the warning information in the fog layer and can also send warning signals to the cloud layer via the WiFi wireless local area network (IEEE 802.11 Standard, WLAN [13]) for driving an alarm system or tripping the haemodialysis machine in a dialysis room. Then, warning information from the personalised physiological monitor can be received on the iPad or smart phone. The experimental results (pig blood) will demonstrate the efficiency of the proposed prototyping model. This new digital analytical and computing model lowers the cost and improves the performance for distributed computing in blood leakage-level identification and multibed monitoring application.

2. Flexible array photosensor: A photocell sensor was made of a high-resistance semiconductor and was a light-controlled variable resistor (illumination: 0.1–1000.0 lx). It tended to be sensitive to light spectrum between 500 nm (green light) and 700 nm (red light) and could act as a dark-activated switching circuit, manipulating the low or high resistors (switch on/off) based on minor blood leakage or blood loss covering any photocell sensor. In this study, four photocell sensors were arranged in an array sensing plane on a flexible substrate, as shown in Fig. 2. The electronic circuits could be printed on a plastic substrate (40×40 mm^2 in size) using the computer numeric control machine and semi-auto screen-printing machine, thus provided flexibility, thinness, and light weight to be easily placed on the puncturing site and for continuous monitoring.

Four resistor voltage dividers were connected to a constant voltage source of V_{cc} = +5.0VDC (current: 0.008–0.5 mA). Then, the sensing nodal voltages, V_i, i = 1, 2, 3, 4, could be obtained to identify the sensing states on the analogue input connectors. The analogue input ports were used to measure the 0.0–5.0VDC voltage signals. Then, a digital sensing signal, s_i, i = 1, 2, 3, 4, as 4-bit binary patterns, could be obtained using the hard limit function with the threshold value, $V_{cc} \times 70\%$ (seen in Fig. 3), as follows:

$$s_i = \begin{cases} 0, & \text{if } V_{cc} \times 70\% < V_i < V_{cc} \\ 1, & \text{if } V_i < V_{cc} \times 70\% \end{cases} \quad (1)$$

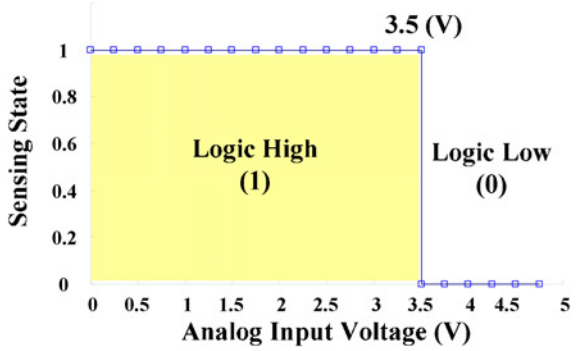

Fig. 3 *Hard limit function*

where sensing state $s_i = 1$ for logic 'High(1)' and $s_i = 0$ for logic 'Low(0)'; threshold valve, $V_{cc} \times 70\%$, was chosen based on at least 40 experimental tests. Sensing state: $\boldsymbol{S} = [s_1, s_2, s_3, s_4] = [0/1, 0/1, 0/1, 0/1]$. For four digit binary numbers, each bit had two states, {0, 1}, and the total combination of the four-digit binary string was $2^4 = 16$ of different binary patterns, and the corresponding binary output patterns, $\boldsymbol{R} = [0/1, 0/1, 0/1]$, could be encoded a binary value of '1', indicating a 'possible event', and everything else encoded by the value of '0'. Three risk levels were indicated as follows:

- Risk Level 1 as $\boldsymbol{R} = [1, 0, 0]$: 0 sensing node could be detected with 1 normal condition, and then first output signal '1' as a high level (+5.0VDC) drove a green LED.
- Risk Level 2 as $\boldsymbol{R} = [0, 1, 0]$: 0 < sensing nodes ≤ 2 could be detected with a total number of 10, and second output signal '1' drove a yellow LED and a loud alarm.
- Risk Level 3 as $R = [0, 0, 1]$: sensing nodes > 2 could be detected with a total number of 5, and third output signal '1' drove a red LED and a loud alarm.

The output pattern, R, was used to identify the possible risk level and to drive the LEDs and the loud alarm units. For 16 binary patterns, an HAM model was used to design a digitised alarm unit to detect the blood leakage/blood loss. Then, warning information was transmitted from the sensing unit to mobile appliances via WiFi wireless synchronous serial communication [13]. For fog computing design, this framework could reduce the communication needed between the sensor and the central data centre by performing analytics and knowledge generations. It facilitated the operation of compute and analysis using the microcomputer and microcontroller [14, 15].

3. HAM machine learning model: An associative memory machine learning method is an unsupervised learning system and can be divided into autoassociative memory and HAM models. The HAM machine learning model is a feedback mechanism that allows for the generation of new patterns, noise filtering, and pattern completion [11, 12]. Its network contains an input layer, an output layer, and network connections. The HAM model can store high-dimensional training patterns in a connecting and associate matrix for modelling human cognitive processes. Its mechanism investigates various conditions that are represented by encoding weighted values in a weighting matrix. This process can maximise information representation and reduce the memory store for each patient with 16 input–output paired training data, as $\boldsymbol{S}_k$–$\boldsymbol{R}_k$, where $k = 1, 2, 3, \ldots, 16$. Hence, it can be used to solve non-linear separable problems. The network connections between the process units are bidirectional and in a loop configuration [16]. This model can learn and recall various types of input and output associations in different data lengths and data types, such as binary, bipolar, and numerical data. The digitised model can deal with binary data and also act as the biological behaviour to perform the associative memory to directly drive the LEDs and the alarm units.

The HAM machine learning algorithm stores information and matrices using noise-free versions of the input and output patterns. To resolve the non-linear separable problem, its configuration can be modified as a multilayer pattern mechanism with non-linear processing units, such as Gaussian functions, as shown in Fig. 4. The HAM algorithm has two stages, the '*learning stage*' and the '*recalling stage*', as delineated below.

Learning stage

Step 1: establish the 16 input–output pairs of the training patterns, $\boldsymbol{S}_k$ and $\boldsymbol{R}_k$, $k = 1, 2, 3, \ldots, 16$.
Step 2: establish the connecting matrix $\boldsymbol{C}$ using K pairs of training patterns

$$\boldsymbol{C} = \sum_{k=1}^{K} \boldsymbol{S}_k^t \boldsymbol{R}_k \tag{2}$$

where $\boldsymbol{C} = [w_{ij}]_{n \times m}$, $\boldsymbol{S}_k = [s_{k1}, \ldots, s_{ki}, \ldots, s_{kn}]^t$, $n = 4$ and $m = 3$, $s_{ki} \in \{0, 1\}$, and $\boldsymbol{R}_k = [r_{k1}, \ldots, r_{kj}, \ldots, r_{km}]^t$, $r_{kj} \in \{0, 1\}$.
Step 3: calculate m eigenvalues

$$\lambda_j = \frac{1}{n} \sum_{i=1}^{n} w_{ij}, \quad j = 1, 2, 3, \ldots, m \tag{3}$$

The weight matrix, $\boldsymbol{W}_{5\times3}$, and the associative matrix, $\boldsymbol{A}_{5\times3}$, for the four risk levels are

$$\boldsymbol{W} \Leftrightarrow \boldsymbol{A}$$
$$\begin{bmatrix} \omega_o \times \lambda_1 & \omega_0 \times \lambda_2 & \omega_0 \times \lambda_3 \\ \omega_1 \times \lambda_1 & \omega_1 \times \lambda_2 & \omega_1 \times \lambda_3 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \omega_4 \times \lambda_1 & \omega_4 \times \lambda_2 & \omega_4 \times \lambda_3 \end{bmatrix} \Leftrightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

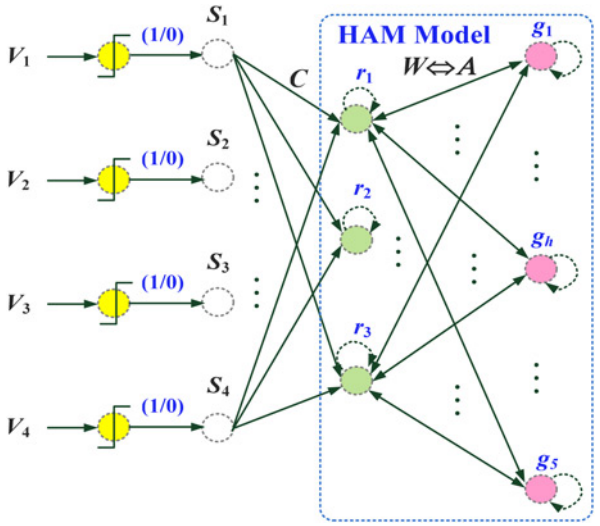

Fig. 4 *Configuration of proposed HAM model*

doi: 10.1049/htl.2017.0091

where $\omega_h = h$, $h = 0, 1, 2, \ldots, 4$, is the weight value for the risk levels. The associative matrix, $\boldsymbol{A}$, is encoded the binary values of 1 or 0, with a value of '1' for a 'possible risk level' and all other states encoded as the value '0'.

Recalling stage

Step 1: obtain the network connecting matrices, $\boldsymbol{C}$, $\boldsymbol{W}$, and $\boldsymbol{A}$, and apply the testing input pattern, $\boldsymbol{S}_0 = [s_1, s_2, s_3, s_4]$, to the connecting network,
Step 2: associate the output pattern, $\boldsymbol{R}_0 = [r_1, r_2, r_3]^{\mathrm{T}}$, as

$$\boldsymbol{R}_0 = \boldsymbol{C}^{\mathrm{T}}\boldsymbol{S}_0, \quad r_j = \sum_{i=1}^{4} w_{ji}s_i, \quad j = 1, 2, 3 \tag{5}$$

Step 3: transit the output pattern, $\boldsymbol{R}_0$, to the Gaussian function units, g_h, and compute the output of the Gaussian function

$$g_h = \exp\left(\frac{-1}{2\sigma^2} \times (\mathrm{ED}_h)^2\right), \quad h = 1, 2, 3, \ldots, 5 \tag{6}$$

$$\mathrm{ED}_h = ||\omega_{hj} - r_j|| = \sqrt{\sum_{j=1}^{3} (\omega_{hj} - r_j)^2} \tag{7}$$

where $\sigma = 0.1$ is the standard deviation; in the vector $\boldsymbol{G}_0 = [g_1, g_2, g_3, \ldots, g_5]$; ED_h is the distance estimation; and the output of Gaussian function, g_h, is the index to screen the similarity degree among the weight values in five row weight vectors. The similarity degree is parameterised with Gaussian function, varying between the values 0 and 1.
Step 4: transit the outputs of g_h units to the r_j unit with nonlinear feedback, and compute the output of r_j unit using the hard limit function with the threshold value 0.50, as

$$v_j = \sum_{h=1}^{5} a_{jh}g_h, \quad j = 1, 2, 3 \tag{8}$$

$$r_j = \begin{cases} 1, & v_j \geq 0.50 \\ 0, & v_j < 0.50 \end{cases}, \quad \boldsymbol{R} = [r_1, r_2, r_3] \tag{9}$$

Step 5: transit the bidirectional patterns repeatedly between the $\boldsymbol{R}_j$ units and g_h units until the bidirectional stability is reached, $v_{\max} = \mathrm{argmax}(v_j) \geq 0.50$ and $\Delta R_p = ||R_p - R_{p-1}|| = 0$, where p is the iteration number. The iteration process (forward and backward computing) takes, $p \leq 2$, iterative computations to reach the convergent condition.

The proposed intelligent algorithm can be easily implemented in Arduino® (Uno, Atmel 8-bit CMOS microcontroller 32 K bytes self-programmable mechanism, six analogue inputs, 14 digital inputs/outputs, DI/digital output (DO)) prototyping platform in the fog layer. The output vector $\boldsymbol{R}$ indicates the digital outputs, $\boldsymbol{R}_p = [r_1, r_2, r_3] = [\text{green, yellow, red}]$, which sets the digital output state as either 'logic high' or 'logic low'. The output LED is active when the digital output is in the high level. Using WiFi wireless communication, a WLAN is used to link to a wearable or a mobile devices in the 2.4-GHz medical frequency band [13], which is used to transmit warning information to nephrology nurses and further an alarm signal to trip the haemodialysis machine in the haemodialysis room (20×30 m^2).

4. Experimental results: The experimental setup for blood leakage detection is shown in Fig. 5. The resistor voltage divider was employed to measure each photoresistor between the DC voltage source and pull-down resistor. Then, four analogue input connectors were used to obtain four nodal voltages from the sensing unit to the four analogue/digital converters (ADCs) as 10-bit ADC channels, as seen in Fig. 6*a*. The analogue voltage reading ranged from 0.0VDC to about 5.0VDC. For animal experiments, pig blood was collected with some anticoagulant to prevent the clotting of blood. This blood sample was heated at 35–37°C to simulate adult blood. We dropped pig blood on the sensing unit in random order using the precision graduated syringe. Thus, pig blood could be used to mock the blood leakage (3–40 ml) with a <1 s reaction time (Fig. 5). When blood (pig blood) leakage covers any photocell sensor, the total

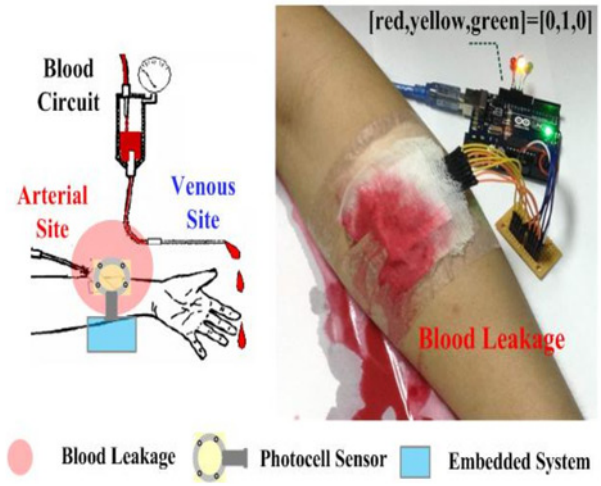

Fig. 5 *Experimental setup for blood leakage detection (pig blood)*

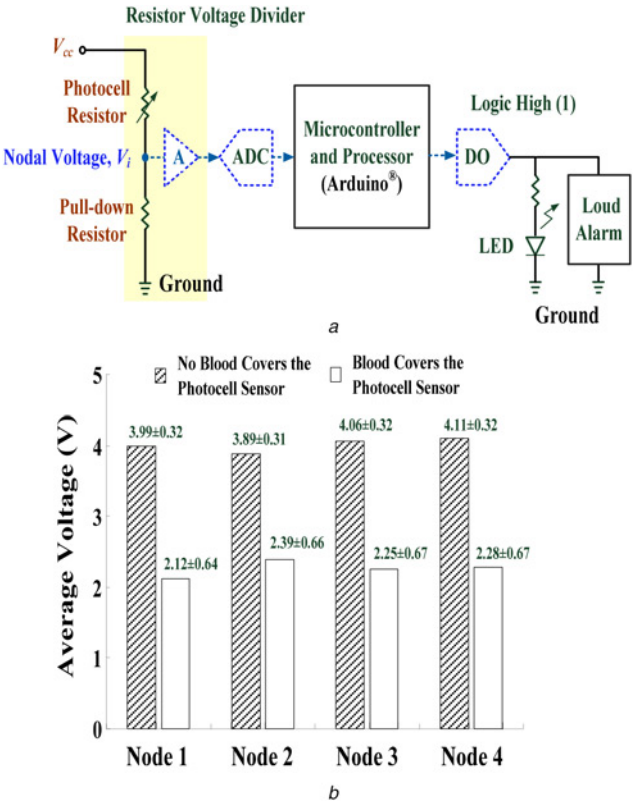

Fig. 6 *Analog input connectors and average nodal voltages*
a Analogue input connectors and DO indication
b Nodal voltage distributions on each photocell sensor

Table 1 Experimental results for blood leakage detection

Risk level	Analogue input (voltage)				Sensing state (0/1)				HAM output	Hit rate %
	V_1	V_2	V_3	V_4	s_1	s_2	s_3	s_4		
1	3.95	3.97	4.12	4.04	0	0	0	0	[1 0 0]	100
2	**2.98**	3.67	4.12	4.08	**1**	0	0	0	[0 1 0]	100
	3.73	**3.05**	4.21	4.16	0	**1**	0	0	[0 1 0]	
	3.35	4.15	**2.52**	4.11	**1**	0	**1**	0	[0 1 0]	
	4.06	4.17	4.13	**2.51**	0	0	0	**1**	[0 1 0]	
	3.44	**3.16**	3.78	4.16	**1**	**1**	0	0	[0 1 0]	
	3.95	**1.82**	**2.67**	4.09	0	**1**	**1**	0	[0 1 0]	
	4.04	4.17	**1.88**	**2.33**	0	0	**1**	**1**	[0 1 0]	
	1.99	3.91	**2.55**	4.07	**1**	0	**1**	0	[0 1 0]	
	1.83	4.03	4.15	**2.54**	**1**	0	0	**1**	[0 1 0]	
	3.78	**3.44**	4.16	**3.16**	0	**1**	0	**1**	[0 1 0]	
3	4.08	**2.03**	**2.10**	**2.39**	0	**1**	**1**	**1**	[0 0 1]	100
	2.10	4.17	**2.42**	**2.51**	**1**	0	**1**	**1**	[0 0 1]	
	2.07	**2.28**	4.16	**2.61**	**1**	**1**	0	**1**	[0 0 1]	
	2.02	**2.30**	**2.53**	4.06	**1**	**1**	**1**	0	[0 0 1]	
	1.57	**2.02**	**2.15**	**2.00**	**1**	**1**	**1**	**1**	[0 0 1]	

resistors of the photocell and the pull-down resistor will increase the current flowing through both resistors will decrease, and thus the voltage across the pull-down resistor will also decrease. The nodal voltage was proportional to the inverse of the photocell resistor. The nodal voltages, V_1 to V_4, could be obtained to identify the sensing states using (1), where the critical threshold, $V_{cc} \times 70\% < V_i < V_{cc}$, $i = 1, 2, 3, 4$, for logic '0', and $V_i < V_{cc} \times 70\%$ for logic '1', respectively.

We can establish the 16 input–output pairs of training patterns, which are two matrices, as the sets of binary patterns. Then, the weight matrix, $\boldsymbol{W}$, and the associate matrix, $\boldsymbol{A}$, could be established using (4), as

$$\boldsymbol{W} = \begin{bmatrix} 0\times0 & 0\times4 & 0\times4 \\ 1\times0 & 1\times4 & 1\times4 \\ 2\times0 & 2\times4 & 2\times4 \\ 3\times0 & 3\times4 & 3\times4 \\ 4\times0 & 4\times4 & 4\times4 \end{bmatrix} \Leftrightarrow \boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Therefore, an HAM machine learning mode had four input nodes, three output nodes, and five non-linear processing nodes (Gaussian function units, as seen in Fig. 4.

For example, considering the blood leakage (>10 ml of pig blood) covered the sensing nodes, 1# and 2#, then, the detection procedure is as shown below.

Step 1: transit the metering analogue voltages, $\boldsymbol{V} = [V_1, V_2, V_3, V_4] = [3.44, 3.16, 3.78, 4.16]$, and estimate the sensing states, $\boldsymbol{S}_0 = [1, 1, 0, 0]$ using the hard limit function (1), as seen in Table 1.
Step 2: initiate the $\boldsymbol{R}_0 = [0, 0, 0, 0]$, then apply the input pattern, $\boldsymbol{S}_0$, and associate the output pattern, $\boldsymbol{R}_0$, using (5).
Step 3: transit the output pattern, $\boldsymbol{R}_0$, to the Gaussian function units, and compute the output vector, $\boldsymbol{G}_0 = [0.00, 0.00, 1.00, 0.00, 0.00]$ and $\mathrm{ED} = [11.31, 5.66, 0.00, 5.66, 11.31]$, using (6) and (7). The maximum one, $g_3 = 1.00$ ($\mathrm{argmin}\|\mathrm{ED}_3\| = 0.00$), is an index to measure the similarity degree among the five categories, then the risk level 2 can be determined.
Step 4: transit the outputs of Gaussian function units, and compute the outputs of v_j, $[v_1, v_2, v_3] = [0.00, 1.00, 0.00]$. The output of r_j units using the hard limit function is $[r_1, r_2, r_3] = [0, 1, 0]$,
Step 5: reach the bidirectional stability and terminate the detection algorithm, and then indicate the 'Risk Level 2#'. Then, output signal r_2 (High Logic) acts to drive a yellow LED and a loud alarm.

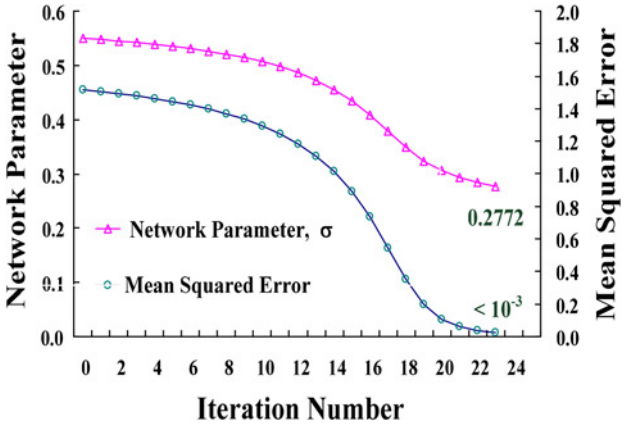

Fig. 7 *Network parameter and mean-squared error versus iteration number for the conventional machine learning model (GRNN)*

When any sensing node was covered by pig blood, the values of nodal voltage changes, V_1 to V_4, were used to identify the sensing states using (1). Fig. 6*b* shows the average nodal voltages by 16 events, including normal condition and possible blood leakage events. Then, the proposed HAM model as a virtual alarm unit was employed to identify blood leakage levels. Hence, the virtual alarm unit could associate two different data types, as a sensing pattern, $\boldsymbol{S}$, and an alarm pattern, $\boldsymbol{R}$, to automatically drive the LED and the loud alarm. Based on at least 80 experimental tests (5×16 events), the experimental results with 16 events indicated a hit rate of 100%, as shown in Table 1. This finding confirmed that the proposed model could detect blood leakage or blood loss during dialysis therapy. Its firmware system integrated with the flexible sensor and the HAM application software took <0.15 ms to deal with the task in the fog layer and further transmitted the warning information via wireless communication to a mobile device or a remote monitor system in the cloud layer.

For the same training patterns, a machine learning model as the generalised regression neural network (GRNN) was also used to establish a screening model with four inputs and three outputs.

Its configuration could be determined using the presentation of 16 input–output pairs of training patterns [17, 18]. We had four input nodes in the input layer, 16 pattern nodes in the pattern layer, four nodes in the summation layer, and three nodes in the output layer (network topology: 4-16-4-3). Its model could deal

doi: 10.1049/htl.2017.0091

Table 2 Comparison of the proposed screening model and GRNN method

Task	Method	
	The proposed screening model	GRNN model
training data	16 input–output pairs of training patterns	16 input–output pairs of training patterns
memory storage	$\boldsymbol{C}$ matrix (4 × 3): 48 bytes $\boldsymbol{W}$ matrix (5 × 3): 60 bytes $\boldsymbol{A}$ matrix (5 × 3): 60 bytes	weight matrix between input and pattern layer (4 × 16): 256 bytes weight matrix between pattern and output layer (16 × 4): 256 bytes
training process	matrix operation	iteration computation<25
recalling process	iteration computation ≤ 2	matrix operation
computer time	<0.15 ms	<5.00 ms
accuracy%	100%	100%

with high-dimensional and non-linear training patterns for prediction and classification applications. However, it needed an optimal algorithm to minimise the pre-specified tolerance value, such as the traditional least-square algorithm or the gradient descent algorithm. The iteration computations were used to reach the convergent condition by tuning the network parameters in the learning stage. The iteration process took about <5.0 ms and <25 iterative computations to reach the convergent condition with the prespecified tolerance value (mean-squared error $\leq 10^{-3}$). Hence, the optimal model parameter, $\sigma = 0.2772$, was obtained to minimise the mean-squared error, as seen in Fig. 7. The GRNN was a regulable pattern mechanism with updating network parameters in online applications. However, initial condition assignments, such as initial network parameters and learning rates, could affect its learning performance. In addition, the fill-in with elements in two weight matrices could increase computing time and memory storage requirements. Considering 4 bytes for digital storage, the memory storage was 512 bytes, as seen in Table 2. Increases in both memory storage and computing time were limited to implementation in a microprocessor-based system or a portable embedded system.

In contrast to the GRNN model, the proposed screening model had a very fast training stage using matrix operation without iteration computations, learning and network parameters assignment. The dimensions of the training pattern presentation and the memory storage needs could be reduced from 512 to 168 bytes. Its recalling stage slightly needed iteration computations without changing any network parameters and took an average execution time of <0.15 ms and ≤2 iterative computations to reach the bidirectional stability. Hence, the HAM model could be easily implemented in an embedded system or a portable detection device. Based on fog computing, the proposed detection model could exert the physical condition, control the measurement process, and send alerts. The wearable assistant tool with the array photocell sensor and the HAM model also reacts to process data and take the decision in the fog layer.

5. Conclusion: The integrating photocell sensors and the associative memory machine learning model were established to detect blood leakage. In contrast to the IR light sensor, the photocell sensor was small, of low cost, consumed low power, and was easy to implement in a wearable device. These cells could be mounted on a flexible printed circuit board such as a plastic substrate. This electrical equipment needs to be validated for safety and effectiveness before its commercialisation by the standard of IEC 60601 series [19] and IEC PAS 63023:2016 [19, 20] that covers the design methodology, verification, and risk assessment. In addition, the proposed model had a simple configuration and a very fast training process in the learning stage and computing process in the recall stage (<0.15 ms and ≤2 iterative computations). The dimensions of the training patterns and the memory storage needs were less than those of the traditional machine learning method. With the high-level programming language (language C), the HAM algorithm could be easily implemented in an intelligent wearable system for personalised physiological monitoring applications. Using pig blood mimicking blood leakage/loss, the experimental results indicated a hit rate of 100% and a true positive rate of 100%. Therefore, the proposed prototype tool could further be integrated into a compact portable microchip without limiting the patient's range of motions. Then, the wireless communication was employed to transit the warning signals to control the alarm system or haemodialysis machine. With its feasibility evaluations, the prototype tool employed the fog (edge) computing and indicated the warning information via wireless communication system for clinical applications in dialysis therapies.

6. Funding and declaration of interests: None declared.

7 References

[1] Axley B., Speranza-Reid J., Williams H.: 'Venous needle dislodgement in patients on hemodialysis', *Nephrol. Nurs. J.*, 2012, **39**, (6), pp. 435–445

[2] Lin C.-H., Chen W.-L., Li C.-M., *ET AL.*: 'Assistive technology using integrated flexible sensor and virtual alarm unit for blood leakage detection during dialysis therapy', *IET Healthc. Technol. Lett.*, 2016, **3**, (4), pp. 290–296

[3] Hurst J.: 'It can happen without warning: venous needle dislodgement', *Renal Bus. Today*, 2009, **4**, (9), pp. 18–22

[4] Sathiya Devi G.S., Joshi A.K.: 'Blood leakage monitoring system using IR sensor in hemodialysis therapy'. Proc. 57th IRF Int. Conf., Pune India, June 2016, pp. 38–42, ISBN: 978-93-86083-35-7

[5] HEMOdialertTM, 2014, Available at https://www.hemodialert.com and www.hemodialert.com/hemodialert-hemodialysis-alarm.php

[6] Perkins L.E.: 'Venous needle dislodgement sensor'. Patent (US 2006/0130591 A1), June 2006, Available at https://www.goole.Ch/patents/US20050038325

[7] Chuang H.-C., Shih C.-Y., Chou C.-H., *ET AL.*: 'The development of a blood leakage monitoring system for the applications in hemodialysis therapy', *IEEE Sens. J.*, 2015, **15**, (3), pp. 1515–1522

[8] Brogi A., Forti S.: 'QoS-aware deployment of IoT applications through the fog'. Technical Reports of the Department of Computer Science, Università di Pisa, March 2016

[9] Skala K., Davidovic D., Afgan E., *ET AL.*: 'Scalable distributed computing hierarchy: cloud, fog, and dew computing', *Open J. Cloud Comput.*, 2015, **2**, (1), pp. 16–24

[10] Gaber M.M., Stahl F., Gomes J.B.: 'Pocket data mining-big data on small devices' (Springer International Publishing, Switzerland, 2014), ISBN 978-3-319-02710-4

[11] Chartier S., Boukadoum M.: 'A bidirectional hetero-associative memory for binary and grey-level patterns', *IEEE Trans. Neural Netw.*, 2006, **17**, (2), pp. 385–396

[12] Chartier S., Giguère G., Langlois D.: 'A new bidirectional heteroassociative memory encompassing correlational, competitive and topological properties', *Neural Netw.*, 2009, **22**, pp. 568–578

[13] Institute of Electrical and Electronics Engineers: 'IEEE std. 802.11-2007, wireless LAN medium access control (MAC) and physical layer (PHY) specifications', 12 June 2007

[14] Bonomi F., Milito R., Zhu J., *ET AL.*: 'Fog computing and its role in the internet of things'. Proc. First Edition of MCC Workshop on Mobile Cloud Computing, 2012, pp. 13–16
[15] Stojmenovic I., Sheng W.: 'The fog computing paradigms: scenarios and security issue'. 2014 Federated Conf. Computer Science and Information Systems, 2014
[16] Tao D., Wen Y., Hong R.: 'Multicolumn bidirectional long short-term memory for mobile devices-based human activity recognition', *IEEE Internet Things J.*, 2016, **3**, (6), pp. 1124–1134
[17] Kan C.-D., Chen W.-L., Lin C.-H., *ET AL.*: 'Optimal flow adjustment of veno-venoarterial extracorporeal membrane oxygenation with an adaptive prediction model: cannula sizes screening and pump speeds estimation', *IET Sci. Meas. Technol.*, 2016, **10**, (3), pp. 177–184
[18] Specht D.F.: 'A general regression neural network', *IEEE Trans. Neural Netw.*, 1991, **2**, (6), pp. 568–576
[19] IEC PAS 63023:2016: Medical electrical system-input interface for haemodialysis equipment for use of external alarming device, Available at https://webstore.iec.ch/publication/24041
[20] IEC 60601-1:2005: International Standard, Medical electrical equipment – Part 1: General requirements for basic safety and essential performance