This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Pyramidal Lucas–Kanade-Based Noncontact Breath Motion Detection

Quoc-Viet Tran, Shun-Feng Su, *Fellow, IEEE*, and Van-Truong Nguyen, *Student Member, IEEE*

***Abstract*—This paper aims to build a simple and low-cost system by using images to detect human breath in a real-time fashion to estimate the peak of the inspiratory phase of a breath so as to define a proper triggering timing for X-ray shooting. In fact, it is very difficult to detect very small breathing motion on images. In this paper, well-known techniques are employed to obtain useful features from the chest area for the Lucas–Kanade algorithm. Various levels of the Pyramidal Lucas–Kanade are then adapted to track possible small motions of those features. The proposed approach can successfully detect the inspiratory–expiratory motions and the peak time of inspiratory phase can be predicted within an acceptable interval of error time. From the experiments conducted, the breath motion can be successfully observed in two different environment situations (dim-lighting and lighting conditions). It can be found that the tracked features are quite robust and stable without losing the quantity over a long period of testing time. Thus, the proposed approach can effectively be used to define a proper triggering timing for X-ray shooting. Besides, in our experiments, even though the target is 6 m away, the breath detection is still successful. In other words, the proposed approach can also be used for surveillance or healthcare environments.**

***Index Terms*—Breath detection, Lucas–Kanade algorithm, noncontact, respiration detection.**

I. Introduction

It IS well-known that the X-ray image can have nice quality if obtained when the patient's ribcage is full of air. Thus, to detect the peak time of an inspiratory phase is important for obtaining good quality X-ray images when taken autonomously. In recent years, many researchers have studied breath detection, which is very helpful in medical diagnostics. Various methods have been proposed for breath detection. Two categories for breath detection approaches can be found; contact and noncontact. Contact-based approaches are to use wearable sensors such as thermistors, ribcage inductive belts [1], [2], pressure sensors [3], [4], wearable systems [5], or gas sensors [6], [7] to directly detect breath motions. Usually, these methods can have accurate results. However, such approaches are not workable for mobile applications or people who are not willing to wear sensors. Noncontact breath detection approaches are to use infrared cameras to measure the temperature on the nasal region [8]–[11], to use microphones near the nose to analyze the acoustic signal [12]–[16], or to use cameras with the temporal differencing technique to find the motion of the patient chest on the vertical direction [17]–[19]. To use CW radars is a simple and cost-effective alternative to detect the frequency change (Doppler effects) of heartbeat and breath [20]–[24]. However, in our case, it is an accessory for an X-ray machine in Hospital. Thus, to have one more radar wave in the environment is not acceptable. Moreover, when the CW radar motion detection mechanism is considered, the radar waves usually directly point to the target. However, the target is moving in a home care environment. Then a target tracking mechanism and moving capability must be added to the CW radar system. In our approach, the breath detecting is conducted by only using images and the results are good and acceptable. Another method is to use a low-power laser irradiating a patient's chest with a smooth reflex surface and let the laser be reflected onto the wall. Then a webcam is used to monitor the laser projection on the wall [25]. There are also studies using Kinects for breath detection. In [26], a Kinect is employed to directly measure morphological changes of the chest wall for respiratory rate detection. However, the accuracy depends on the depth accuracy, which is not very good in Kinect. Besides, patients must wear tight t-shirts to avoid detection errors. In [27], Chest wall motion analysis is conducted by using four Kinects to form three-dimensional representations of a subject's torso over time. It can be seen that this approach has the same limitations as those in [26]. In [28], another method is proposed. It first uses facial detection algorithm by Viola and Jones [29] to detect the mouth region. In order to detect breath motions, the mean value on the mouth region is filtered by infinite impulse response band-pass filter to obtain the frequency range of breath. Then the Eulerian method is employed to obtain the mean value on the mouth region. This mean value is then analyzed to identify the breathing signal. It can be expected that the motion signal on the mouth region is too intricate. In fact, in our analysis, the signal obtained in the mouth area is complicated and is not easy to identify the breath motion as shown in this paper reported later. Thus, in that approach, a depth

Manuscript received December 13, 2017; revised March 4, 2018; accepted April 4, 2018. This work was supported in part by the Swissray Digital X-ray Center, Taiwan and in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 104-2221-E-011-090-MY3 and Grant 104-2811-E-011-012. This paper was recommended by Associate Editor Q. Wang. *(Corresponding author: Shun-Feng Su.)*

Q.-V. Tran and S.-F. Su are with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan (e-mail: quocviet09clc@gmail.com; sfsu@mail.ntust.edu.tw).

V.-T. Nguyen is with the Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan, and also with the Department of Mechanical Engineering, Hanoi University of Industry, Hanoi 159999, Vietnam (e-mail: qvtruongcdt@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2825458

sensor is also used to measure the distance from Kinect to chest region to detect the breath signal. This depth signal is used to compare with the breathing signal obtained from the mouth region. It can be seen that this approach is complicated and also need to use the depth information to detect breath motions. Hence, the accuracy of this part depends on the depth accuracy, which is not very good, especially the breath motion is very small. In our approach, the Lagrangian method (optical flow) is used to track robust features inside the chest region over the time. The results are very good.

In this paper, the change of features in video images is detected by using spatial decomposition. However, the change of image parameters for a breath motion is very difficult to observe directly. Therefore, the resulting signal must be amplified similar to that used in detecting the flow of blood and the small motion for heartbeat in [30]. In the approach [31], features on the face are obtained through some image processing techniques and ways of identifying heartbeats based on the trajectories obtained by the Lucas–Kanade algorithm [34]–[37] is proposed. Such an algorithm is also considered in our approach to detect small motions on the chest area. With the similar idea in [31], this paper is to build a simple and low-cost system to detect human breath in a real-time fashion by using a Microsoft Kinect or even a simple webcam. In this paper, image processing techniques are developed to obtain useful features. The depth information obtained by the Kinect is only employed to remove the background and to create a mask for the RGB image so as to locate chest region for the Lucas–Kanade algorithm. In this paper, when only a normal webcam is used (without depth information), the facial detection algorithm by Viola-Jones [29] can then be applied to identify the chest region. In our approach, after the chest region is obtained, the corners detection [38], [39] is employed to identify features for tracking. Since the original Lucas-Kanade [36] uses the optical flow technique and affine transformation to track features over consecutive frames, it is only applied for a small displacement of pixels. Therefore, the number of lost features may increase while tracking lasts for a long period. This can be seen from our experiments shown later. Thus, the Pyramidal Lucas-Kanade [37] is considered in our approach. With the proposed approach, the breath motion can easily be observed in actual experiments. With those observed breath motion, a way of detecting the peak of the inspiratory phase of a breath from images is proposed to define a proper timing for trigging X-ray shooting. The experimental results show the effectiveness of the proposed system.

The outline of this paper is stated as follows. After this introduction, the whole proposed system is described in Section II. Ways of defining features in images are discussed in Section III. In Section IV, with the use of the Lucas–Kanade algorithm, a novel approach of getting breath motions is presented. In Section V, how to detect the peak times of inspiratory phases is discussed. Various experiments are conducted to demonstrate the effectiveness of the proposed approach. Their results are reported in Section VI. Finally, conclusive remarks are given.

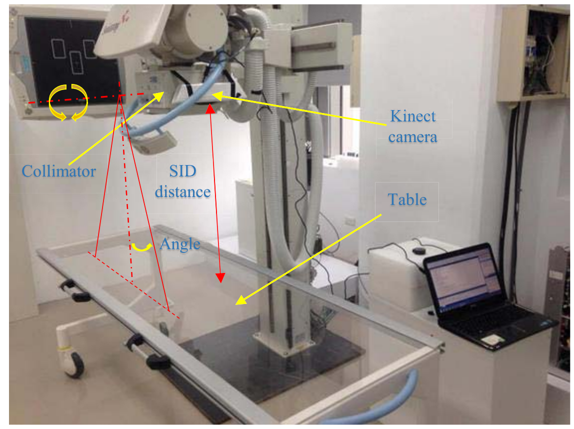

Fig. 1. Operational environment of the system.

II. System Description

This paper is to design a simple and low-cost system to detect human breath so as to help in triggering X-ray shooting autonomously. The developed system is simple and only requires a Microsoft Kinect or a normal webcam without any complicated setup or multiple devices. A Kinect is mounted on a ddRCruze X-ray machine, a product of the Swissray company, as shown in Fig. 1. With the Kinect, this X-ray system has been equipped with many features [55] such as calibration angle for collimator and table, measuring the distance from collimator to table, etc. For the economical issue, the available Kinect is used to estimate the peak times of inspiratory phases of a patient in this paper. In fact, a normal webcam can also be used to detect the breath in our approach.

Visual C++ foundation is used to interact with Kinect camera to capture RGB and depth images. It includes OpenNI library that supports the Kinect camera and the OpenCV library that supports for computer vision applications. In this implementation, the Microsoft Visual Studio C++ in Window 10 Pro environment is used.

The algorithm of the proposed breath detection system is illustrated in Fig. 2. In our approach, feature points are selected to be tracked by the Pyramidal Lucas–Kanade algorithm [37]. In our implementation, there are two modes for defining those feature points. In the manual mode, the feature points are created by clicking designated points on the region by the user. Auto mode is to define those features through image processing techniques automatically. Detailed algorithms will be given in the next section. From our experiments, as shown later, the movements of breath are significant enough to be observed. In this paper, a way of estimating the peak time of inspiration is proposed. The details of the above processes are introduced in the following sections.

III. Feature Extraction

In this section, the approaches used for processing the images obtained from the Kinect are presented. Besides, the corner detection used for defining features is also presented in this section.

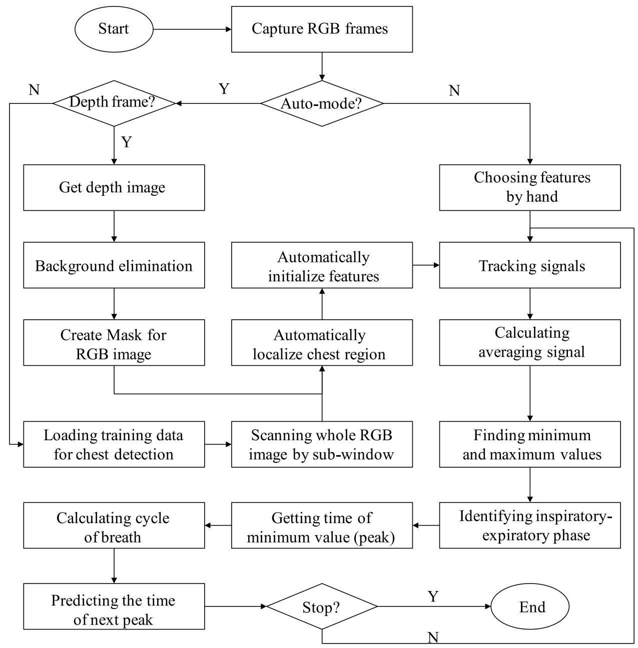

Fig. 2. Algorithm for breath detection.

Fig. 3. Depth image (left) after removing background.

Fig. 4. Depth image (left) after dilation.

A. Chest Allocation Using Depth Sensor

In defining feature points, the first step is to remove the background of the image obtained. It is easy to do that with a Kinect device [43]–[47]. Two thresholds (upper and lower) are used to consider pixels outside this interval being background pixels. The result of an example is shown in Fig. 3. However, from the example, it can be observed that the quality of the depth image is not good at the edge of the object because some depth information is lost. To enhance the quality of depth image, the usual dilation morphology approach [40]–[42] is employed. The result after dilation as shown in Fig. 4 is much better than that in Fig. 3.

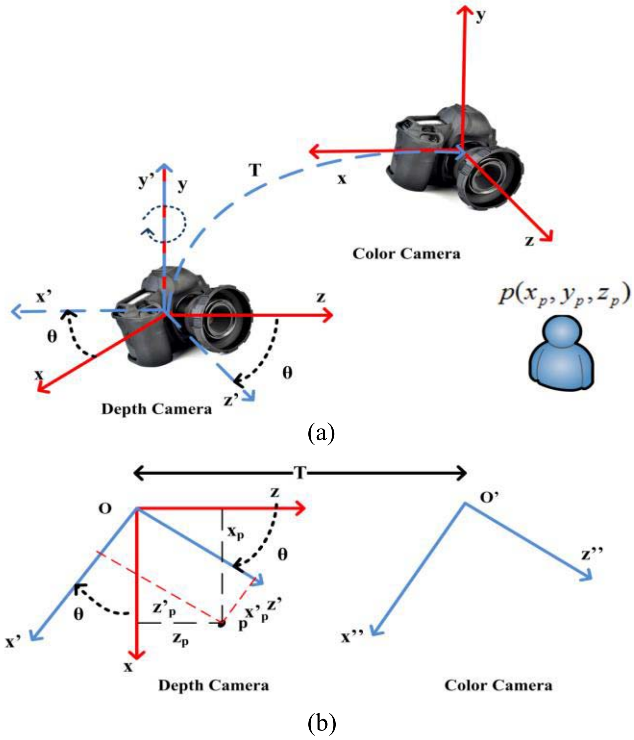

Fig. 5. Geometric model of RGB-D camera. (a) RGB-D camera overview. (b) Geometric model of Depth and Color Camera under $x-z$ plane projection.

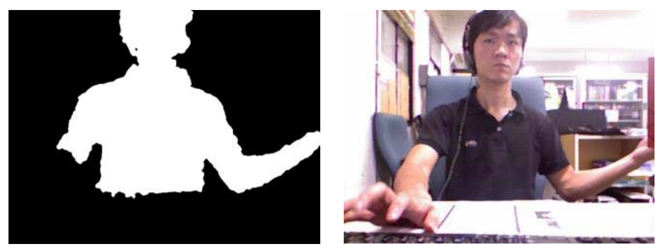

Fig. 6. Depth (left) and RGB image (right) before registration.

The Kinect has two cameras: one for capturing depth information and another for capturing RGB images. Since these two cameras have different coordinates, it is necessary to make calibration [54] to overlap two images perfectly. Let an object captured by the depth camera have the coordinate $p(x_p,\ y_p\ ,z_p)$. To align the coordinate of the depth camera to that of the RGB camera, a transformation as shown in Fig. 5 is used. First, the coordinate for the depth camera is rotated around the y-axis with an angle θ and the new coordinate is

$$\begin{pmatrix} x'_p \\ z'_p \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_p \\ z_p \end{pmatrix}. \tag{1}$$

Note that the coordinate for the y-axis is not changed. Next, if the difference from the depth coordinate to the color coordinate is $T(t_x,\ t_y,\ t_z)$, the transformation is

$$\begin{aligned} x''_p &= x'_p + t_x \\ z''_p &= z'_p + t_z. \end{aligned} \tag{2}$$

t_y is not used because two cameras have the same norm vector on the y-axis. From Figs. 6 and 7, it can be seen the color

Fig. 7. Depth (left) and RGB image (right) after registration.

Fig. 8. Mask of depth image and RGB image.

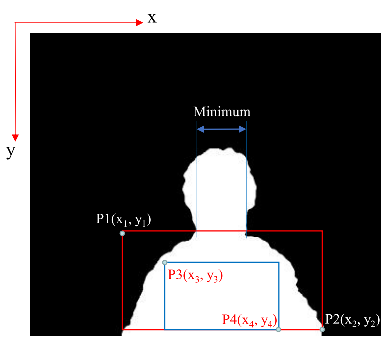

Fig. 9. ROI of chest.

image is matched with the depth information after making registration.

After removing background and location registration, the body segmentation is conducted. An example is shown in Fig. 8. As shown in the later discussion, the chest region can provide more obvious movement signal. The body segmentation is to find the chest region. First, the body area can be identified by two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ as shown in Fig. 9, where x_1 and x_2 are obtained by finding the smallest and the largest positions of pixels for the foreground object on the x-direction. For the location on the y-axis, the first step is to find the neck area, which is the narrow region of the foreground object on the y-axis and larger than a calculated threshold value

$$\text{threshold} = \frac{f.d}{\text{SID}} = \frac{w.d}{2\text{SID}.\tan(\text{FoV}/2)} \tag{3}$$

where f, w, and FoV are the focal length, screen width, and field of view angle of the Kinect camera, respectively. d is the real width of patient's neck and SID is the minimum distance from Kinect to the body part as shown in Fig. 10(b). Thus, the histogram along the y-axis [Fig. 10(a)] is obtained and as shown in Fig. 9, y_1 is the point corresponding to the minimum value of the histogram that is over the threshold.

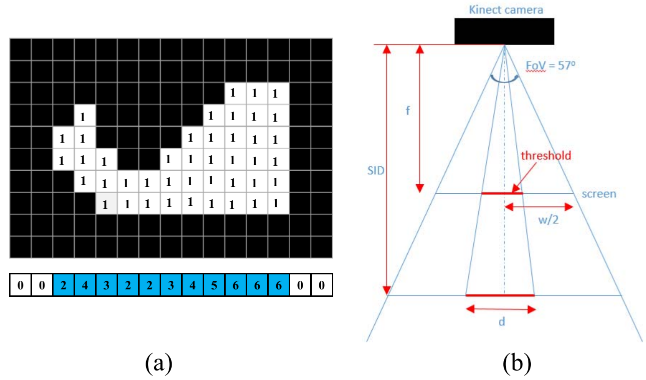

Fig. 10. (a) Projection on y-axis. (b) Threshold to find accuracy neck region.

TABLE I
CONFUSION TABLE

Actual/predicted	Chest	Non-Chest	Total	Recall (%)
Chest	455	31	486	93.621
Non-Chest	42	472	514	91.829
Total	497	503	1000	
Precision (%)	91.549	93.837		**Acc**: 92.7%

For the lower boundary, from common body structure, in our implementation, y_2 is simply defined to be

$$y_2 = y_1 + 2(x_2 - x_1)/3. \tag{4}$$

As shown in Fig. 9, the rectangle region constructed by $P1(x1,\ y1)$ and $P2(x2,\ y2)$ is the body region and is obtained directly from the foreground image. From the figure, it can be found that this region may contain some areas not belong to the foreground. In order to avoid getting features in those improper areas, the original region is shrunk and a smaller region of interest (ROI) is created with two points $P_3(x_3,\ y_3)$ and $P_4(x_4,\ y_4)$ as shown in Fig. 9, where

$$x_3 = x_1 + (x_2 - x_1)/5 \tag{5}$$
$$y_3 = y_1 + (y_2 - y_1)/4 \tag{6}$$
$$x_4 = x_1 + 4(x_2 - x_1)/5 \tag{7}$$
$$y_4 = y_2. \tag{8}$$

These formulas are to define the chest area without any background region. It is a simple heuristic process and as long as the final ROI obtained is completely within the chest area, it is acceptable.

B. Chest Allocation Using Machine Learning in RGB Images

The system developed in this paper uses the available Kinect to allocate the chest region in the automatic mode. In this paper, the case of using a normal webcam is also considered (i.e., no depth information is available). Several image processing techniques can be considered to allocate the chest region in the automatic mode. The algorithm used in [29] is used and the overall process is shown in Fig. 11.

In our implementation, 1000 positive images and 2000 negative images are used in the training process. First, features are extracted by applying a set of Haar features as shown in Fig. 12. The AdaBoost is then used to collect important features and to train the classifier. Most significant features are

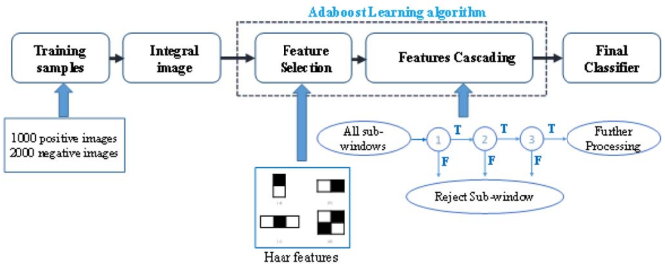

Fig. 11. Overall training process for chest detection.

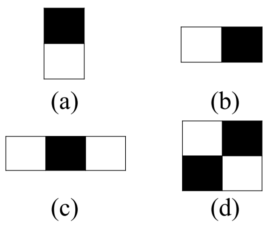

Fig. 12. Two-rectangle features are shown in (a) and (b). (c) Three-rectangle feature and (d) four-rectangle feature.

selected from a huge number of features in thousands of negative and positive samples by the AdaBoost. A single classifier is not sufficient to achieve the real-time constraints for object detection. Therefore, strong classifiers are constructed into a cascade of classifiers which helps to increase the detection performance and to reduce the computational time. A cascade of classifiers can train high dimensional feature space with a high accuracy. In this paper, 15 cascade classifiers are used for training data. As shown in Table I, 1000 images are considered as the test data set. It can be found that the performance is good and the algorithm can be implemented for real-time chest detection in our system.

C. Corner Detection

In our approach, corner detection [38], [39] is employed to define features points for tracking. Corner features are identified by analyzing the distribution of gradients over a window to distinguish whether it is a flat region, an edge or a corner. In the approach, the following matrix is formed:

$$M = \begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{pmatrix} \tag{9}$$

where I_x and I_y are the partial derivatives of the image I with respect to the x-axis and the y-axis, respectively. The predominant directions and magnitudes of gradients within the window are encoded by two eigenvalues λ_1 and λ_2 of the matrix. As stated in [38] and [39], corner points will have large values for both λ_1 and λ_2 as shown in Fig. 13. Thus, corners are the spots with the corner quality measure $R = \min(|\lambda_1|, |\lambda_2|)$ over a predefined threshold (λ). To determine λ, Jianbo and Tomasi [38] first selected a region that has the approximate brightness to measure the eigenvalues to define a lower bound for λ. Then, various kinds of features and highly textured regions are selected to obtain an upper bound for λ. In their experiments, those two bounds are comfortably separated and the value λ, chosen halfway in-between, is not critical.

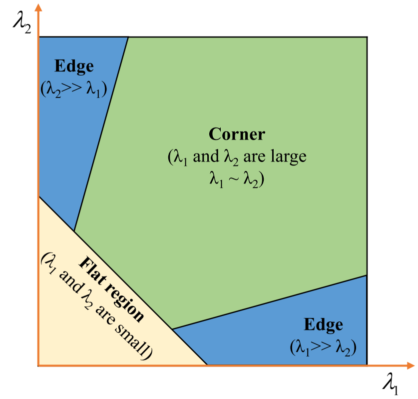

Fig. 13. Classification of image points using eigenvalues of *M*.

Therefore, the upper bound for λ is chosen to identify corners in this paper. The results shown in our experiments later indicate such a selection is robust and stable for those features selected. The results of using corner detections are shown in Section VI. From those results, it can be observed that the corners are robust and stable.

IV. MOTION DETECTION

Various approaches have been implemented for motion detection such as using wearable micro inertial sensors to estimate the upper limb motion and head motion [48], [49], or using an overhead stereo system to track people motion [50], or using multiple single-row laser-range scanners to track pedestrians [51]. Usually, these methods can have accurate results as reported in the literature. However, these approaches are not workable for a small motion by cameras or sensors from a significant distance. As mentioned, the Lucas–Kanade method is employed for motion detection in this paper. The basic idea of optical flow [52], [53] is that the brightness of a feature point in two consecutive frames must be the same. Thus, it is to find feature pairs in two consecutive frames satisfying the following equation:

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \tag{10}$$

$$\Leftrightarrow I(x, y, t) = I(x, y, t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt \tag{11}$$

$$\Rightarrow \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt = 0 \tag{12}$$

$$\Rightarrow I_x dx + I_y dy + I_t dt = 0 \tag{13}$$

$$\Rightarrow I_x u + I_y v = -I_t. \tag{14}$$

In the Lucas–Kanade method [34]–[36], the motion of features between two consecutive frames is supposed to be small and approximately constant within the neighborhood of a point under consideration. Therefore, a $n \times n$ window is considered to solve the optical flow equation (14) so as to find out the local image flow vector (u, v) as

$$\begin{aligned} I_x(x_1, y_1, t)u + I_y(x_1, y_1, t)v &= -I_t(x_1, y_1, t) \\ I_x(x_2, y_2, t)u + I_y(x_2, y_2, t)v &= -I_t(x_2, y_2, t) \\ &\vdots \\ I_x(x_n, y_n, t)u + I_y(x_n, y_n, t)v &= -I_t(x_n, y_n, t) \end{aligned} \tag{15}$$

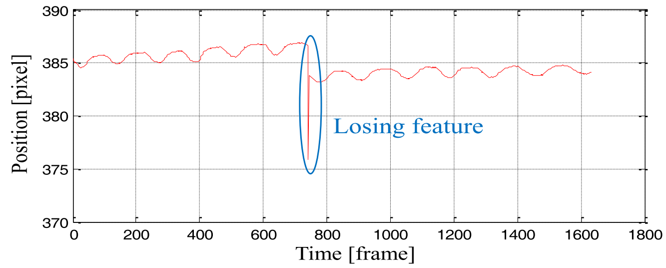

Fig. 14. Losing weak features over the time.

where $I_x(x_i, y_i, t)$, $I_y(x_i, y_i, t)$, and $I_t(x_i, y_i, t)$ are the partial derivatives of the image I with respect to position x_i, y_i, and the time t, respectively. The matrix form of (15) can be written as $AX = b$, where

$$A = \begin{bmatrix} I_x(x_1, y_1, t) & I_y(x_1, y_1, t) \\ I_x(x_2, y_2, t) & I_y(x_2, y_2, t) \\ \vdots & \vdots \\ I_x(x_n, y_n, t) & I_y(x_n, y_n, t) \end{bmatrix}$$

$$X = \begin{bmatrix} u \\ v \end{bmatrix} \text{ and } b = \begin{bmatrix} -I_t(x_1, y_1, t) \\ -I_t(x_2, y_2, t) \\ \vdots \\ -I_t(x_n, y_n, t) \end{bmatrix}. \tag{16}$$

Therefore, the two unknown variables (u, v) can be solved by the least squares principle

$$A^T AX = A^T b \text{ or } X = \left(A^T A\right)^{-1} A^T b \tag{17}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} I_x(x_i, y_i, t)^2 & \sum_{i=1}^{n} I_x(x_i, y_i, t) I_y(x_i, y_i, t) \\ \sum_{i=1}^{n} I_y(x_i, y_i, t) I_x(x_i, y_i, t) & \sum_{i=1}^{n} I_y(x_i, y_i, t)^2 \end{bmatrix}^{-1} \times \begin{bmatrix} -\sum_{i=1}^{n} I_x(x_i, y_i, t) I_t(x_i, y_i, t) \\ -\sum_{i=1}^{n} I_y(x_i, y_i, t) I_t(x_i, y_i, t) \end{bmatrix}. \tag{18}$$

The original Lucas-Kanade [36] is used for tracking small motions. However, features may be lost significantly when there is a large motion as shown in Fig. 14, which is the tracking history in one breath detection case. In our application, the breath motion is very small. However, patients may change their posture suddenly due to discomfort in some cases. That causes large motions. Thus, the detection may not be robust if the original Lucas–Kanade is used. A modified approach called the Pyramidal Lucas–Kanade has been proposed in [37]. Fig. 15 demonstrates three-level pyramids of two frames. For a given feature point u on image I, its corresponding location $v = u + d$ is found on image J, where $d = [d_x \; d_y]$ is the displacement from u to v. The local affine transformation matrix between image I and image J is then created from the vicinity of u and v, respectively. Supposed L is the level of pyramid image. For m levels, $L = 0, \ldots, L_m$, define $u^L = \left[u_x^L \quad u_y^L\right]$, and then the corresponding coordinates of the point u on the pyramidal image I^L are computed as

$$u^L = \frac{u}{2^L}. \tag{19}$$

The overall process for tracking features by using the Pyramidal Lucas–Kanade [37] is implemented as follows: first, the optical flow and affine transformation matrix is computed from the corresponding features between two continuous frames at the deepest pyramid level (L_m). The computational results are then propagated to its upper-level $L = L_m - 1$ to form initial features (u^L, v^L) and the affine transformation matrix. The refined optical flow and the affine transformation is computed at level $L = L_m - 1$ and then propagated to level $L = L_m - 2$. The process is iterated up to level 0 (the original image). As shown in Fig. 15, a large displacement ($d_x^{L=0} = 16$ pixels) become a small displacement in level 3 ($d_x^{L=3} = 2$ pixels). Therefore, the Pyramidal Lucas–Kanade [37] with three levels of pyramids and (31 × 31) size of the search windows at each pyramidal level is used to chase possible matched points on the consecutive frames in this paper.

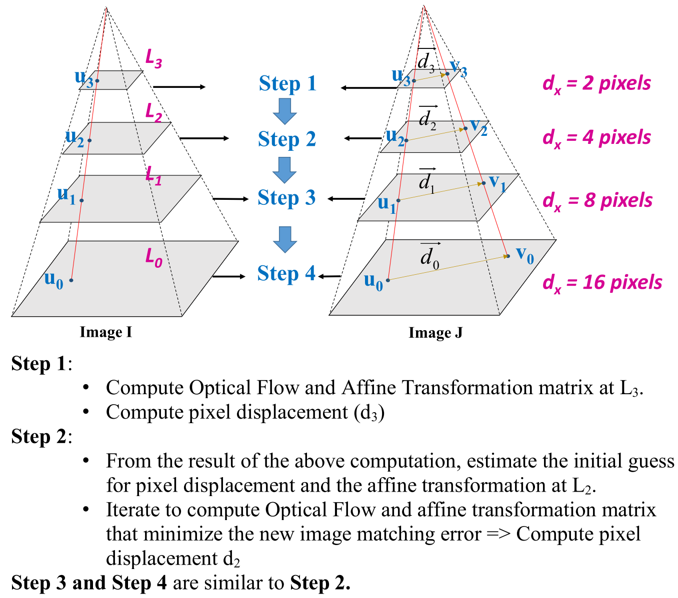

Fig. 15. Three-level pyramids of two continuous frames.

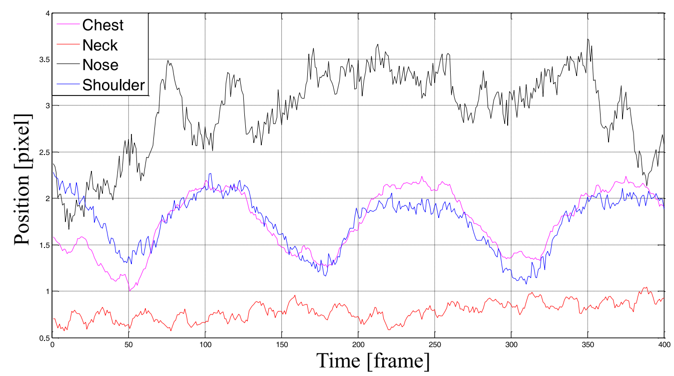

Fig. 16. Signals in different regions.

In this paper, various regions of an image have been considered in the breath motion detection and the signals obtained are shown in Fig. 16. The unit of the time axis is frames because all information is associated with image frames. From the results, it is obvious that the breath motions cannot be observed from the neck and nose regions. It seems reasonable because there is no Newtonian reaction for breath like

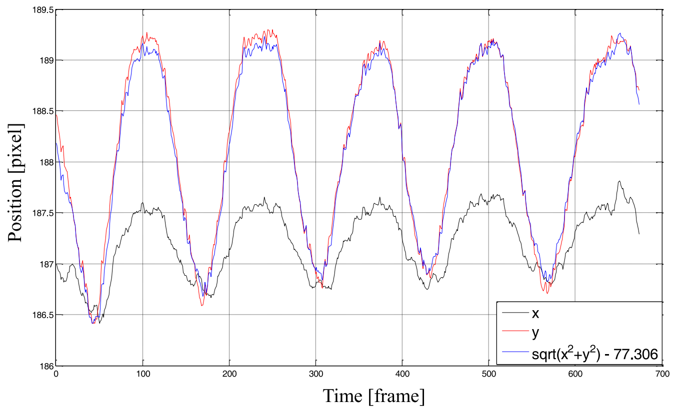

Fig. 17. *X*-direction, *Y*-direction and shifted combined signals.

heartbeat and there are various other motions in these two regions. The chest and the shoulder regions can have clear motion effects. In this paper, sometimes, when a patient lies down and wears loose clothes, the features points on the shoulder may have difficult to track breath motions. Therefore, in this paper, the chest region is considered.

Note that in the above we only consider motions on the y-axis. In our implementation, both axes are considered and their results are shown in Fig. 17. From the results, it is evident that the movement (roughly 2.5 pixels fluctuation) on the vertical (the y-axis) signal is better than that (roughly 1-pixel fluctuation) of the horizontal (the x-axis) signal. The combination of x and y signals obtained as

$$\text{signal} = \sqrt{x^2 + y^2} \tag{20}$$

is also considered. As shown in Fig. 17, the fluctuation of the vertical signal is quite similar to the combined signal. It is reasonable because the fluctuation magnitude of the y-axis is more 2.5 times than that of the x-axis in the experiment. In this paper, the patient is fixed on the table of ddRCruze X-ray machine following the vertical direction. Therefore, in order to reduce computation burden, only vertical signal is chosen to analyze and to detect the breath. From our experiments, it can be observed that the noncontact breath detection is very successful.

V. Peak Estimation

Next, the peak time of a breath cycle is estimated from the obtained signal. As shown in Fig. 18, the minimum and the maximum values correspond to the time of the deepest inspiration and expiration, respectively. A Boolean variable "Status" is created to identify the phase of respiration. The Status is set to false or true if the signal gets the maximum or minimum value, respectively. It can be found that the inspiration phase corresponds to the false and the expiration phase is true. Supposed that t_2 is the current time as shown in Fig. 18. The cycle can easily be obtained by subtracting the previous peak time (t_1) from the current peak time (t_2)

$$\text{cycle} = t_2 - t_1. \tag{21}$$

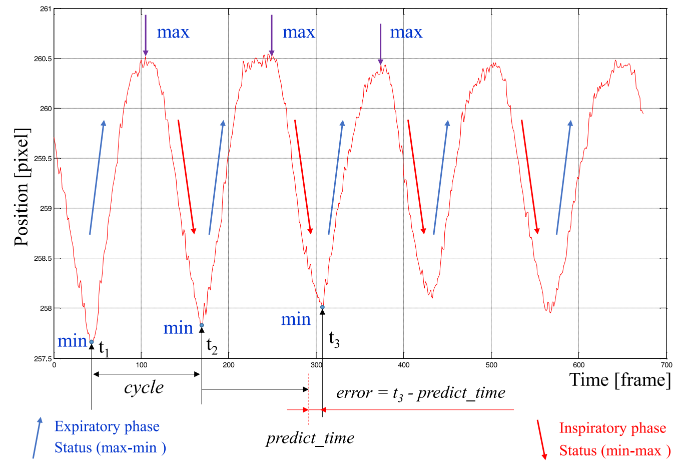

Fig. 18. Identify inspiratory–expiratory phase.

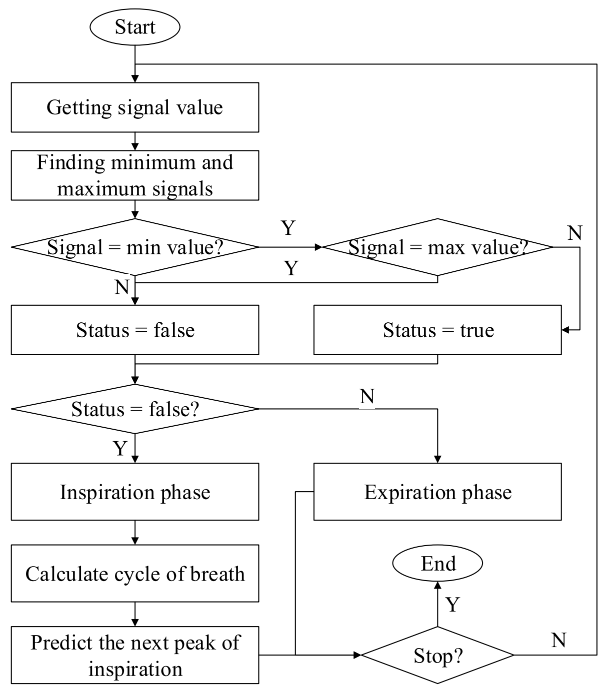

Fig. 19. Detecting phase of inspiration and expiration.

The time of the next peak is simply predicted as

$$\text{predict\_time} = t_2 + \text{cycle}. \tag{22}$$

The flow chart of the peak detection is shown in Fig. 19.

Another method is to get the average value of many previous cycles to predict the next peak time. This method is aimed at reducing the error between the actual and the predicted value. As shown in Fig. 20, t_0 is the current time and t_i $(i = 1, \ldots, n)$ are the previous peak times. The previous cycles of breath are calculated as

$$\text{cycle}_i = t_{i-1} - t_i \ \forall (i = 1 \cdots n). \tag{23}$$

The averaging cycle is calculated as

$$\text{avg\_cycle} = \sum_{i=1}^{n} \text{cycle}_i / n. \tag{24}$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

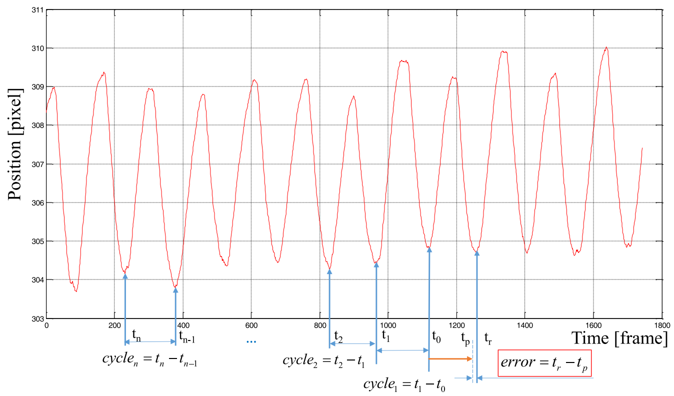

Fig. 20. Calculate previous cycles of breath and predict the time of deepest inspiration.

TABLE II
PROCESSING TIMES AND NUMBER OF FEATURE POINTS FOR USING LUCAS–KANADE

Number of features	Core i5			Core i7		
	Processing time [ms]	Percentage of increase [%]	Mean of peak error [ms]	Processing time [ms]	Percentage of increase [%]	Mean of peak error [ms]
1	37.5	0	192.188	33.878	0	294.468
20	40.5	8	87.764	33.921	0.127	404.440
40	40.6	8.267	152.250	34.052	0.514	402.427
60	40.7	8.533	190.964	34.181	0.894	427.363
80	42.3	12.8	247.751	34.247	1.089	315.244
100	42.7	13.867	256.200	34.300	1.246	345.641

The time of the next peak is predicted as

$$t_p = t_0 + \text{avg\_cycle}. \tag{25}$$

VI. EXPERIMENTAL RESULTS

In this session, the results of various experiments for verifying the effectiveness of the proposed ideas are reported.

A. Processing Time for Different Numbers of Feature Points

In this part, various cases are tested to verify the influence of different numbers of feature points on the processing time. Manual mode is considered in these cases. First, different numbers of feature points (1, 20, 40, 60, 80, 100) are created on the chest regions. The Lucas–Kanade algorithm [36] is then used for tracking these features. The average processing times of a frame over one minute for two different CPUs are tabulated in Table II. It can be found that the peak errors are independent to the number of features. Also, the processing time increases only 13.867% when the number of features increases from 1 to 100 when using Core i5, while only 1.246% for Core i7. It is evident that the number of feature points have little influence on the processing time, especially for Core i7 PC. Sometimes, in a detection process, the feature points may be lost as shown in Fig. 14. As mentioned, the Pyramidal Lucas–Kanade [37] is applied to overcome possible feature loss in the original Lucas–Kanade. To compare the tracking performance between the original Lucas–Kanade and the Pyramidal Lucas–Kanade, many large motions have been recorded for analyzing the number of lost features. From Fig. 21, it is clearly evident that when more layers of the pyramid are used, the performance of tracking features is better. Therefore, in this paper, three-level of the pyramid is preferred for tracking features on the chest region. Since pyramid technique is used for tracking features, the processing time may increase compared to that of the original Lucas–Kanade. Table III shows the processing time between different levels of pyramid and number of feature points. As shown in Table III, the processing time is slightly related to the number of pyramid levels and the number of feature points, but not significant. Even with 100 feature points, the processing time increases only 10.01% for the three-level pyramid. However, the feature points are more robust for tracking large motions with the Pyramidal Lucas–Kanade [37].

TABLE III
PROCESSING TIMES AND NUMBER OF FEATURE POINTS FOR USING PYRAMIDAL LUCAS–KANADE

Number of feature points	Processing time [ms] of number of pyramidal levels			
	Level = 0	Level = 1	Level = 2	Level = 3
1	33.3399	33.5700	34.7866	34.9951
20	33.4561	33.6672	34.8563	35.2090
40	33.6924	33.8522	35.3241	35.8332
60	33.7283	33.8831	35.3486	36.1239
80	33.8670	34.1267	35.5831	36.3893
100	33.9305	34.4151	35.8212	37.3622

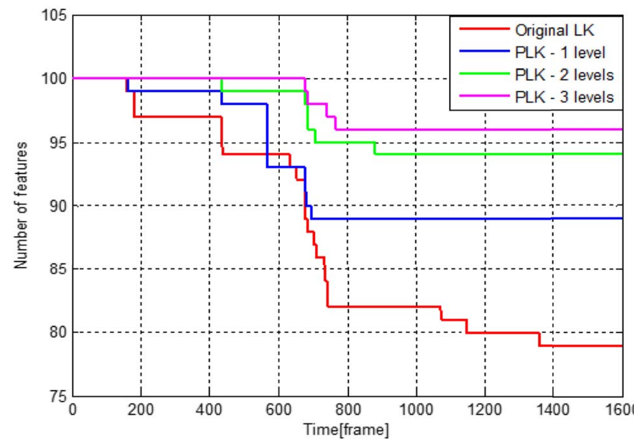

Fig. 21. Tracking performance with large motions.

Fig. 22. Five features are initialized on the chest region.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE IV
STATISTIC OF FLUCTUATED TIME IN ONE MINUTE

	0-4 [Frame]	5-9 [Frame]	10-14 [Frame]	15-30 [Frame]	Processing Time [ms]
P1	5	3	4	1	34.277
P2	2	5	6	0	34.665
P3	5	3	3	1	34.860
P4	3	2	3	4	34.932
P5	0	3	7	2	35.432
Sum	15	16	23	8	
Average					34.833

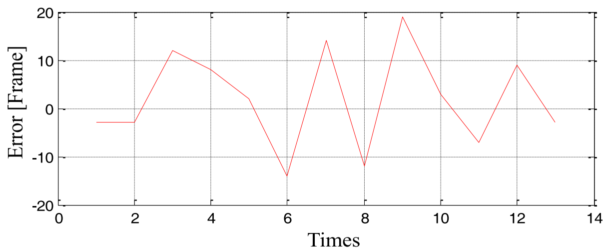

Fig. 23. Fluctuation between predicted and actual peak time.

B. Analysis of the Errors for Predicted Peaks of Inspiration

In this section, data of breath motions are collected from different persons to analyze the fluctuation between the predicted and the actual peak time of inspiration. As shown in Fig. 22, five feature points are selected on the chest region to get the signals of breath. First, the simple prediction based on the previous cycle is employed. There are five persons considered in our experiments, the error for the predicted peak in one minute is summarized in Table IV, and the detailed errors for the first target are given in Fig. 23. From our consultation with radiologists, when the breath peak prediction error is less than one second, it is good enough to have good quality of x-ray images. In our analysis, the predicted errors are mostly less than 0.5 second. As shown in Fig. 24, all testing images and predicted time errors are satisfactory from the radiologists viewpoint.

C. Environment Effects Analysis

Different environments are considered to test the effectiveness of the proposed system. The period of five minutes is considered. A traditional 110 cm of distance (SID) is tested. The results are shown in Fig. 25 and its error history is shown in Fig. 26. Fig. 27 shows the error statistics in a different level of bins. From the result, it is evident that the breath can be observed without any problem. Then, the number of feature points in the process is concerned in different distances under different lighting conditions. In our implementation, for the lighting condition, all lamps are turned on and for the dim-lighting condition, a half of lamps are turned off. The obtained images are shown in Fig. 28. Different SID distances from 70 cm to 110 cm are also considered in these two conditions during 5 min. The results are shown in Table V. It can be found that the number features are stable. That means the corner detection algorithm by calculating the minimal eigenvalue of gradient matrices

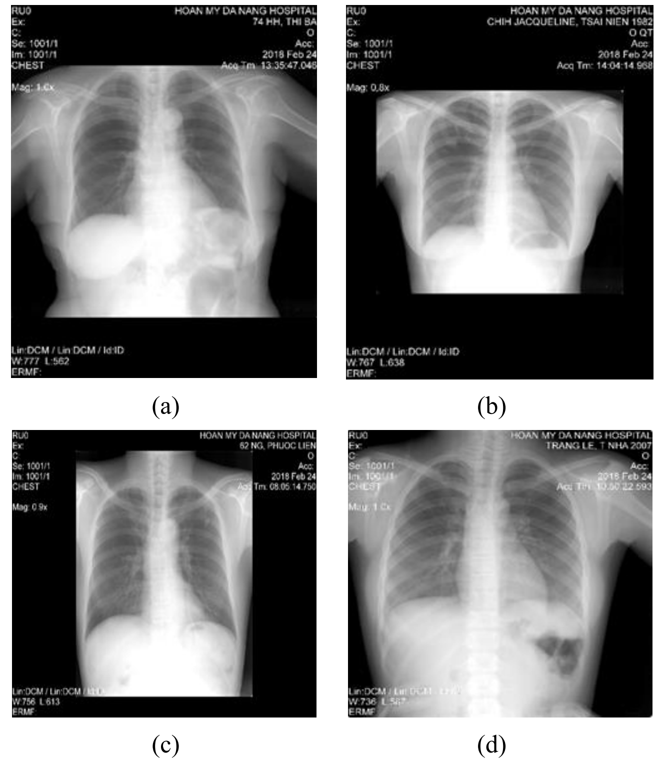

Fig. 24. X-ray image with corresponding predicted errors. (a) Predicted error = +0.372 s. (b) Predicted error = +0.137 s. (c) Predicted error = +0.281 s. (d) Predicted error = −0.229 s.

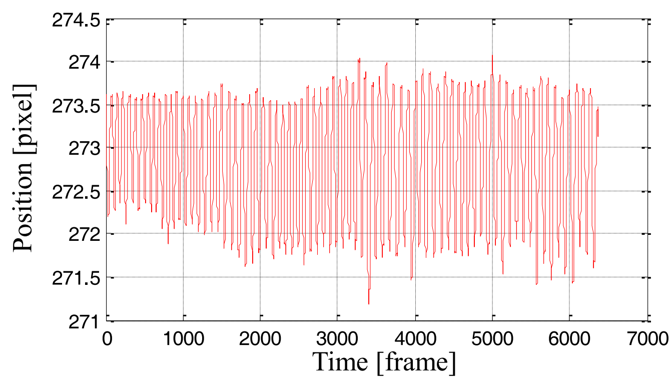

Fig. 25. Averaging signal of 120 feature points in 5 min—SID 110 cm.

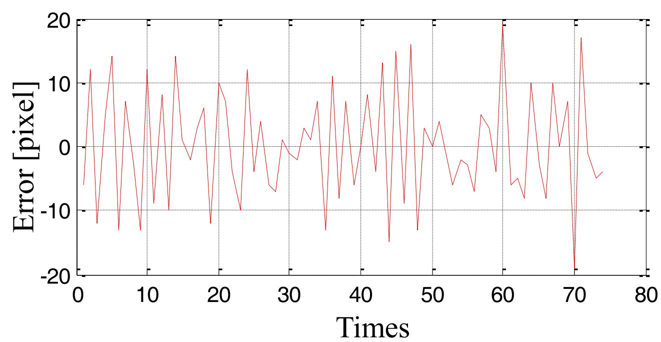

Fig. 26. Fluctuation between predicted and actual peak time—SID 110 cm.

that combines with the Pyramidal Lucas–Kanade [37] is very good to determine the strong corner features for tracking on an image.

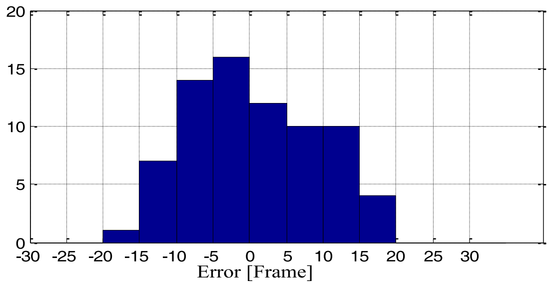

Fig. 27. Histogram of fluctuated peak time in 5 min—SID 110 cm.

Fig. 28. Dim lighting (left) and lighting (right) environments.

TABLE V
DATA SID DISTANCES AND NUMBER OF FEATURES IN TWO DIFFERENT LIGHTING CONDITIONS

SID [cm]	Number of feature points			
	Dim lighting		Lighting	
	Starting time	Ending time	Starting time	Ending time
70	120	120	120	120
80	120	120	120	120
90	120	120	120	120
100	120	120	120	120
110	120	120	120	120

In addition to the use for taking X-ray, the proposed approach is also tested for surveillance or healthcare environment. Different distances from 1 m to 6 m are considered to check the capability of the feature tracking. In these cases, five features are chosen to test during one minute and the results are shown in Fig. 29. It is clearly evident that the motions of breath can easily be observed in all six cases even on the horizontal direction, on which the motion fluctuation is less significant as stated in above. Of course, when the distance is 6 m, the obtained signal is with more noise. Nevertheless, the breath movement can be well detected. The signal of breath fluctuates around 1.4 pixels in a 1-m distance and decreases to 0.15 pixels in the case with a 6-m distance. It indicates that the Pyramidal Lucas–Kanade algorithm tracks small motions very well.

D. Peak Time Prediction With More Cycles

In this part, the peak prediction by using more previous cycles is studied. The signals and the errors of peak times are recorded by getting an averaging cycle of 1∼5 previous cycles in one and five minutes, respectively. The error statistics are shown in Table VI.

From Table VI, in a short period (one-minute case), the error seems to be reduced when more previous cycles are used, but not so significantly as expected. When the breath during a long time (5-min case) is considered, the errors do not get better even worse when more previous cycles are used. From Fig. 30, it can be observed that the widths of those cycles may change. It is because the breath cycle can be changed intentionally and is not constant. Nevertheless, in all cases, almost the errors fluctuate within 10 frames. In other words, this result is acceptable for real-time applications.

TABLE VI
DATA OF PEAK ERRORS WITH NUMBER OF PREVIOUS CYCLES

Number of previous Cycles	Error in one minute [frame]			Error in 5 minutes [frame]		
	Min	Max	Mean	Min	Max	Mean
1	1	20	8.9	0	24	7.368
2	0	16	6	0	23	6.131
3	0	16	5.7	1	25	8.254
4	0	15	7.538	0	29	9.020
5	1	11	5.667	0	24	8.906

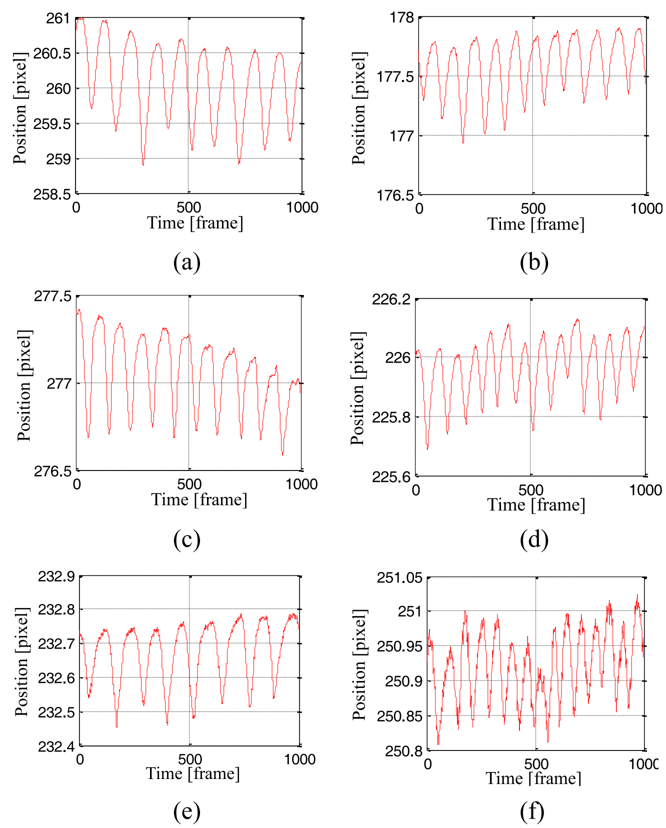

Fig. 29. Signal in different distances following horizontal direction. (a) One meter. (b) Two meters. (c) Three meters. (d) Four meters. (e) Five meters. (f) Six meters.

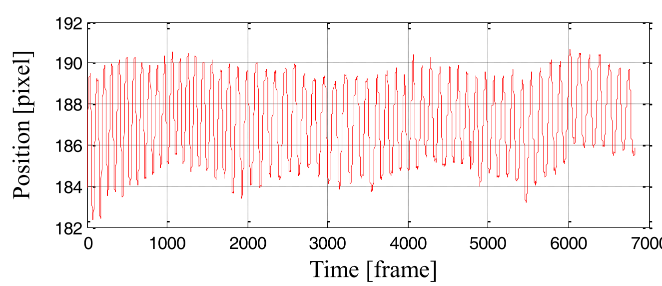

Fig. 30. Averaging signal of breath during 5 min.

VII. Conclusion

In this paper, a novel noncontact approach using images is proposed to detect breath motions. From the experiments conducted, the breath motions can easily be observed to detect the inspiratory–expiratory phase, the time of peak inspiration, and the breath rate. Our algorithm is tested under various cases to see the fluctuated time between the actual and the predicted time. Also, the effects of a number of feature points to the processing time are tested. From our experiments, the errors of the predicted times of a peak inspiration fluctuate around 0.5 s. It is acceptable for the case of trigging X-ray shooting. Moreover, the features are stable and robust with different distances and lighting environments. From our experiments, the signal of breath fluctuates in 1.4 pixels in a 1-m distance and 0.15 pixels in a 6-m distance for a normal breath. In other words, those breath motions can be observed significantly even in a long distance up to 6 m. Thus, this technique can be employed in surveillance or healthcare systems. Further research will be implemented to detect the breath of babies. Since babies do not lie statically as adults do while measuring breath. Therefore, the movement of breath signal is a combination of many motions. It is expected that a band-pass filter is required to extract the frequency range that corresponds to baby's breath motions.

References

[1] K. P. Cohen, J. G. Webster, J. Northern, Y. H. Hu, and W. J. Tompkins, "Breath detection using fuzzy sets and sensor fusion," in *Proc. 16th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. Eng. Adv. New Opportunities Biomed. Eng.*, vol. 2, 1994, pp. 1067–1068.

[2] K. P. Cohen, Y. H. Hu, W. J. Tompkins, and J. G. Webster, "Breath detection using a fuzzy neural network and sensor fusion," in *Proc. Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 5. Detroit, MI, USA, 1995, pp. 3491–3494.

[3] Y.-W. Liu and Y.-L. Hsu, "Development of a bed-centered telehealth system based on a motion-sensing mattress," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, Manchester, U.K., 2013, pp. 1466–1470.

[4] S. Lokavee, T. Puntheeranurak, T. Kerdcharoen, N. Watthanwisuth, and A. Tuantranont, "Sensor pillow and bed sheet system: Unconstrained monitoring of respiration rate and posture movements during sleep," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, Seoul, South Korea, 2012, pp. 1564–1568.

[5] C. D. Katsis, N. Katertsidis, G. Ganiatsas, and D. I. Fotiadis, "Toward emotion recognition in car-racing drivers: A biosignal processing approach," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 3, pp. 502–512, May 2008.

[6] C. M. Ionescu, J. A. T. Machado, and R. D. Keyser, "Analysis of the respiratory dynamics during normal breathing by means of pseudophase plots and pressure–volume loops," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 1, pp. 53–62, Jan. 2013.

[7] C. M. Ionescu, "From viscoelastic models to lung function devices," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, 2016, pp. 002533–002540.

[8] D. Hanawa *et al.*, "Nasal cavity detection in facial thermal image for non-contact measurement of breathing," in *Proc. 35th Int. Conf. Telecommun. Signal Process. (TSP)*, 2012, pp. 586–590.

[9] D. Hanawa *et al.*, "Nose detection in far infrared image for non-contact measurement of breathing," in *Proc. IEEE-EMBS Int. Conf. Biomed. Health Inform.*, 2012, pp. 878–881.

[10] J. Gwak, M. Shino, K. Ueda, and M. Kamata, "Effects of changes in the thermal factor on arousal level and thermal comfort," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, 2015, pp. 923–928.

[11] S. L. Bennett, R. Goubran, and F. Knoefel, "The detection of breathing behavior using Eulerian-enhanced thermal video," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2015, pp. 7474–7477.

[12] O. Yahya and M. Faezipour, "Automatic detection and classification of acoustic breathing cycles," in *Proc. Zone 1st Conf. Amer. Soc. Eng. Educ. (ASEE Zone)*, Bridgeport, CT, USA, 2014, pp. 1–5.

[13] T. Rosenwein, E. Dafna, A. Tarasiuk, and Y. Zigel, "Breath-by-breath detection of apneic events for OSA severity estimation using non-contact audio recordings," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2015, pp. 7688–7691.

[14] M. Igras and B. Ziólko, "Wavelet method for breath detection in audio signals," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, San Jose, CA, USA, 2013, pp. 1–6.

[15] T. Rosenwein, E. Dafna, A. Tarasiuk, and Y. Zigel, "Detection of breathing sounds during sleep using non-contact audio recordings," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Chicago, IL, USA, 2014, pp. 1489–1492.

[16] Y. Nishida, T. Hori, T. Suehiro, and S. Hirai, "Monitoring of breath sound under daily environment by ceiling dome microphone," in *Proc. IEEE Int. Conf. Syst. Man. Cybern. (SMC)*, vol. 3. Nashville, TN, USA, 2000, pp. 1822–1829.

[17] Y.-W. Bai, W.-T. Li, and Y.-W. Chen, "Design and implementation of an embedded monitor system for detection of a patient's breath by double webcams in the dark," in *Proc. 12th IEEE Int. Conf. e-Health Netw. Appl. Services (Healthcom)*, 2010, pp. 93–98.

[18] Y.-W. Bai, W.-T. Li, and Y.-W. Chen, "Design and implementation of an embedded monitor system for detection of a patient's breath by double webcams," in *Proc. IEEE Int. Workshop Med. Meas. Appl. (MeMeA)*, Ottawa, ON, Canada, 2010, pp. 171–176.

[19] Y.-W. Bai, W.-T. Li, and C.-H. Yeh, "Design and implementation of an embedded monitor system for body breath detection by using image processing methods," in *Dig. Tech. Papers Int. Conf. Consum. Electron. (ICCE)*, Las Vegas, NV, USA, 2010, pp. 193–194.

[20] J. J. Szczyrek and W. Winiecki, "On detection and estimation of breath parameters using ultrawideband radar," in *Proc. IEEE 8th Int. Conf. Intell. Data Acquisition Adv. Comput. Syst. Technol. Appl. (IDAACS)*, 2015, pp. 18–21.

[21] M. Pieraccini, G. Luzi, D. Dei, L. Pieri, and C. Atzeni, "Detection of breathing and heartbeat through snow using a microwave transceiver," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 1, pp. 57–59, Jan. 2008.

[22] W. Li, B. Tan, and R. J. Piechocki, "Non-contact breathing detection using passive radar," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2016, pp. 1–6.

[23] Y.-L. Hsu, M.-C. Chen, C.-M. Cheng, and C.-H. Wu, "Development of a portable device for home monitoring of snoring," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, vol. 3. Waikoloa, HI, USA, 2005, pp. 2420–2424.

[24] İ. Şişman, A. O. Canbaz, and K. Yeğin "Micro-doppler radar for human breathing and heartbeat detection," in *Proc. Comput. Electromagn. Int. Workshop (CEM)*, 2015, pp. 1–2.

[25] Y.-W. Bai, Y.-W. Chen, and W.-T. Li, "Design of an embedded monitor system with a low-power laser projection for the detection of a patient's breath," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Las Vegas, NV, USA, 2011, pp. 553–554.

[26] F. Benetazzo, A. Freddi, A. Monteriù, and S. Longhi, "Respiratory rate detection algorithm based on RGB-D camera: Theoretical background and experimental results," *Healthcare Technol. Lett.*, vol. 1, no. 3, pp. 81–86, 2014.

[27] J. M. Harte *et al.*, "Chest wall motion analysis in healthy volunteers and adults with cystic fibrosis using a novel Kinect-based motion tracking system," *Med. Biol. Eng. Comput.*, vol. 54, no. 11, pp. 1631–1640, 2016.

[28] A. Procházka, M. Schätz, O. Vyšata, and M. Vališ, "Microsoft Kinect visual and depth sensors for breathing and heart rate analysis," *Sensors*, vol. 16, no. 7, p. 996, 2016.

[29] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1. Kauai, Hawaii, USA, 2001, pp. 511–518.

[30] H.-Y. Wu *et al.*, "Eulerian video magnification for revealing subtle changes in the world," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–8, 2012.

[31] G. Balakrishnan, F. Durand, and J. Guttag, "Detecting pulse from head motions in video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, 2013, pp. 3430–3437.

[32] Q. V. Tran, S.-F. Su, and M.-C. Chen, "Breath detection for enhancing quality of *X*-ray image," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, 2016, pp. 1–4.

[33] Q.-V. Tran, S.-F. Su, C.-C. Chuang, V.-T. Nguyen, and N.-Q. Nguyen, "Real-time non-contact breath detection from video using AdaBoost and Lucas-Kanade algorithm," in *Proc. Joint 17th World Congr. Int. Fuzzy Syst. Assoc. 9th Int. Conf. Soft Comput. Intell. Syst. (IFSA-SCIS)*, 2017, pp. 1–4.

[34] B. D. Lucas, "Generalized image matching by the method of differences," Ph.D. dissertation, Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Jul. 1984.

[35] J. Y. Bouguet, *Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the Algorithm*, Res. Labs, Santa Clara, CA, USA, 2001.

[36] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. 7th Int. Joint Conf. Artif. Intell.*, vol. 2. Vancouver, BC, Canada, 1981, pp. 121–130.

[37] J. Y. Bouguet, *Pyramidal Implementation of the Affine Lucas Kanade Feature Tracker Description of the Algorithm*, Intel Corporat., Santa Clara, CA, USA, 2001, pp. 1–10.

[38] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Seattle, WA, USA, 1994, pp. 593–600.

[39] C. Tomasi and T. Kanade, "Detection and tracking of point features," Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Rep. CMU-CS-91-132, Apr. 1991.

[40] J. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: Academic Press, 1983.

[41] J. Serra, "Image analysis and mathematical morphology," in *Theoretical Advances*, vol. 2. London, U.K.: Academic Press, 1988.

[42] E. R. Dougherty, *An Introduction to Morphological Image Processing*. Bellingham, WA, USA: SPIE Optical Eng. Press, 1992.

[43] A. Sobral and A. Vacavant, "A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos," *Comput. Vis. Image Understand.*, vol. 122, pp. 4–21, May 2014.

[44] T. Bouwmans, J. Gonzàlez, C. Shan, M. Piccardi, and L. Davis, "Special issue on background modeling for foreground detection in real-world dynamic scenes," *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1101–1103, 2014.

[45] M. Camplani, C. R. del Blanco, L. Salgado, F. Jaureguizar, and N. García, "Advanced background modeling with RGB-D sensors through classifiers combination and inter-frame foreground prediction," *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1197–1210, 2014.

[46] E. J. Fernandez-Sanchez, L. Rubio, J. Diaz, and E. Ros, "Background subtraction model based on color and depth cues," *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1211–1225, 2014.

[47] M. Narayana, A. Hanson, and E. G. Learned-Miller, "Background subtraction: Separating the modeling and the inference," *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1163–1174, 2014.

[48] Z.-Q. Zhang, L.-Y. Ji, Z.-P. Huang, and J.-K. Wu, "Adaptive information fusion for human upper limb movement estimation," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 5, pp. 1100–1108, Sep. 2012.

[49] S. Polak, Y. Barniv, and Y. Baram, "Head motion anticipation for virtual-environment applications using kinematics and EMG energy," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 569–576, May 2006.

[50] J. García, A. Gardel, I. Bravo, J. L. Lázaro, and M. Martínez, "Tracking people motion based on extended condensation algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 3, pp. 606–618, May 2013.

[51] H. Zhao and R. Shibasaki, "A novel system for tracking pedestrians using multiple single-row laser-range scanners," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 2, pp. 283–291, Mar. 2005.

[52] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM Comput. Surveys*, vol. 27, no. 3, pp. 433–466, 1995.

[53] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput. Vis.*, vol. 12, no. 1, pp. 43–77, 1994.

[54] S. Dinç *et al.*, "Depth-color image registration for 3D surface texture construction using Kinect camera system," in *Proc. IEEE SOUTHEASTCON*, Lexington, KY, USA, 2014, pp. 1–6.

[55] P.-R. Zhan, S.-F. Su, and M.-C. Chen, "Foreground extraction method for *X*-ray imaging auxiliary system," *New Trends Syst. Sci. Eng.*, vol. 276, pp. 137–145, 2015.

Quoc-Viet Tran received the B.S. degree in automatic production engineering from the Danang University of Science and Technology, Da Nang, Vietnam, in 2014 and the M.S. degree in automation and control from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2016, where he is currently pursuing the Ph.D. degree in system engineering with the Department of Electrical Engineering.

His current research interests include computer vision applications, machine learning, image processing, and intelligent control systems.

Shun-Feng Su (S'89–M'91–SM'05–F'10) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1983 and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1989 and 1991, respectively.

He is currently a Chair Professor with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei. He has published over 200 refereed journal and conference papers in the areas of robotics, intelligent control, fuzzy systems, neural networks, and nonderivative optimization. His current research interests include computational intelligence, machine learning, virtual reality simulation, intelligent transportation systems, smart home, robotics, and intelligent control.

Dr. Su is very active in various international/domestic professional societies. He is currently the President of the International Fuzzy Systems Association. He is also currently in the Boards of Governors of IEEE Systems, Man, and Cybernetics Society and acts as the Young Professionals Subcommittee Chair. He also serves as a Board Member of various academic societies. He also acted as the general chair, the program chair, or various positions for many international and domestic conferences. He currently serves as an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON FUZZY SYSTEMS, and IEEE ACCESS, a Subject Editor (Electrical Engineering) of the *Journal of the Chinese Institute of Engineers*, and the Editor-in-Chief of the *International Journal of Fuzzy Systems*. He is a CACS fellow.

Van-Truong Nguyen (S'17) received the B.S. and M.S. degrees in mechatronics engineering from the Department of Applied Mechanics, School of Mechanical Engineering, Hanoi University of Science and Technology, Hanoi, Vietnam, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree in control system engineering with the Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan.

From 2013 to 2015, he was a Lecturer with the Department of Mechanical Engineering, Hanoi University of Industry, Hanoi. His current research interests include neural networks, intelligent systems, and adaptive controls.