Sleep Apnea Detection via Depth Video & Audio Feature Learning

Cheng Yang *Student Member, IEEE*, Gene Cheung *Senior Member, IEEE*, Vladimir Stankovic *Senior Member, IEEE*, Kevin Chan, Nobutaka Ono *Senior Member, IEEE*

Abstract—**Obstructive sleep apnea, characterized by repetitive obstruction in the upper airway during sleep, is a common sleep disorder that could significantly compromise sleep quality and quality of life in general. The obstructive respiratory events can be detected by attended in-laboratory or unattended ambulatory sleep studies. Such studies require many attachments to a patient's body to track respiratory and physiological changes, which can be uncomfortable and compromise the patient's sleep quality. In this paper, we propose to record depth video and audio of a patient using a Microsoft Kinect camera during his/her sleep, and extract relevant features to correlate with obstructive respiratory events scored manually by a scientific officer based on data collected by Philips system Alice6 LDxS that is commonly used in sleep clinics. Specifically, we first propose an alternating-frame video recording scheme, where different 8 of the 11 available bits in captured depth images are extracted at different instants for H.264 video encoding. At the decoder, the uncoded 3 bits in each frame can be recovered via block-based search. Next, we perform temporal denoising on the decoded depth video using a motion vector graph smoothness prior, so that undesirable flickering can be removed without blurring sharp edges. Given the denoised depth video, we track a patient's chest and abdominal movements based on a dual-ellipse model. Finally, we extract ellipse model features via a wavelet packet transform (WPT), extract audio features via non-negative matrix factorization (NMF), and insert them as input to a classifier to detect respiratory events. Experimental results show first that our depth video compression scheme outperforms a competitor that records only the 8 most significant bits. Second, we show that our graph-based temporal denoising scheme reduces the flickering effect without over-smoothing. Third, we show that using our extracted depth video and audio features, our trained classifiers can deduce respiratory events scored manually based on data collected by system Alice6 LDxS with high accuracy.**

I. Introduction

It is well understood that quantity and quality of sleep could significantly affect work productivity [1]. In particular, *obstructive sleep apnoea*, characterized by repetitive obstruction in the upper airway during sleep, is common in the general population [2] and can have significant negative effect on a person's sleep quality, and hence quality of life and cognitive functions. The condition is diagnosed via attended (in-laboratory) or unattended (ambulatory) diagnostic sleep studies. We address the problem of identifying the obstructive respiratory events in this paper.

To detect different respiratory events (that characterize obstructive apnea, hypopnea and central apnea), there exist in-laboratory monitoring devices such as system Alice6 LDxS (Philips) that measure a patient's physiological parameters such as oxyhemoglobin saturation, oronasal airflow etc, using various sensors physically attached to the patient's body. In particular, according to the American Academy of Sleep Medicine (AASM) Manual 2007 [3], an apnea is defined by a drop in the peak respiratory airflow by ≥ 90% from the baseline and the duration of the event lasts at least 10 seconds. An obstructive apnea is associated with continued or increased inspiratory effort throughout the entire period of absent airflow. In contrast, a central apnea is associated with absent inspiratory effort. A mixed apnea is associated with the initial portion of the event with absent effort followed by the resumption of such in the latter part of the event. A hypopnea is defined by a drop of ≥ 30% airflow from the baseline and the event lasts for at least 10 seconds, and such change is associated with a 4% drop in oxyhemoglobin desaturations [3].

However, existing in-laboratory monitoring devices are cumbersome to use, expensive, and intrusive with multiple body straps and tubes that affect a patient's sleep quality during monitoring. On the other hand, less intrusive sleep monitoring units such as vibration-sensing wristbands (*e.g.*, Fitbit[1] and Jawbone UP[2]) mostly record sleep time, *i.e.*, the *quantity* rather than the *quality* of sleep, and are not equipped to detect respiratory events of different kinds as previously described during the night.

Motivated by the shortcomings of in-laboratory monitoring devices and consumer-level sleep monitoring units, our goal is to accurately but non-intrusively detect respiratory events as manually scored by a scientific officer based on data collected by system Alice6 LDxS. Towards this goal, we propose a *completely contact-less* sleep monitoring system based on depth video and

C. Yang and V. Stankovic are with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, UK, G1 1XQ, (e-mail: {cheng.yang,vladimir.stankovic}@strath.ac.uk).

G. Cheung and N. Ono are with National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyoda-ku, Tokyo, Japan 101–8430 (e-mail: {cheung, onono}@nii.ac.jp).

K. Chan is with Campbelltown and Camden Hospitals, Sydney, Australia, South West Local Area Health Service, New South Wales, Australia and School of Medicine, University of Western Sydney, Australia (e-mail: drkevinchan@yahoo.com.au).

[1] http://www.fitbit.com/

[2] https//jawbone.com/up/

audio processing, suitable for home use. Not relying on the lighting condition of a dark sleeping room, we use a Microsoft (MS) Kinect sensor projecting infrared light patterns to capture depth images of the sleep patient. The main contributions of the paper are as follows:

1) To facilitate transmission of the acquired data from the patient's home to a sleep clinic for storage and analysis, we propose an alternating-frame video recording scheme, so that different 8 of the 11 bits in captured depth images are extracted at different instants for efficient encoding using H.264 Advanced Video Coding (AVC) video codec [4]. At the decoder, the uncoded 3 bits in each frame can be recovered accurately via block-based search.
2) We perform temporal denoising using a motion vector smoothness prior [5], [6] expressed in the graph-signal domain [7], so that unwanted flickering can be removed without blurring sharp edges in the depth images.
3) Given the denoised depth video, we track the patient's chest and abdominal movements over time based on our proposed dual-ellipse model.
4) We extract ellipse model features via a wavelet packet transform (WPT) [8], [9], which are combined with audio features extracted via non-negative matrix factorization (NMF) [10]–[13] for a Support Vector Machine (SVM) and feed-forward neural network (NN) classifiers to detect respiratory events.

Using respiratory events scored by a scientific officer (who is blind to our study) based on data collected by system Alice6 LDxS as ground truth, and captured depth video and audio of patients collected at Concord Private Hospital, Australia, we conducted extensive experiments to test our system. First, we show that our depth video compression scheme outperforms a competitor that records only the eight most significant bits in peak signal-to-noise ratio (PSNR). Second, we show that graph-based temporal denoising scheme reduces the flickering effect without over-smoothing. Third, we show that our system can deduce respiratory events of different kinds as scored manually by a scientific officer based on data collected by system Alice6 LdxS with high accuracy[3].

The outline of the paper is as follows. We first discuss related work and give an overview of our sleep monitoring system in Sections II and III, respectively. We then discuss the components of our system, depth video recording, denoising, ellipse modeling, and feature extraction and classification, in Sections IV, V, VI and VII, respectively. Finally, we present experimental results and conclude this paper in Sections VIII and IX, respectively.

II. Related Work

A. Sleep Monitoring Systems

Recent advances in wireless sensing and multimedia processing have led to the development of many novel sleep monitoring systems, using a variety of sensors such as force, temperature, audio, and image sensors. Most of these systems, however, require wearable sensors (hence not contact-less) or do not have sufficient precision necessary for clinical applications. Numerous smartphone-based systems for sleep disorder detection have emerged recently (see Table 1 in [14]), based on audio recording and accelerometer measurements. However, there is no scientific evidence regarding clinical usability of these systems [14]. Other recent methods not reviewed in [14] are either limited to measuring respiration rate (such as [15]) and sleep duration (*e.g.*, [16]), or require wearable sensors [17], [18]. For example, the system of [17] is capable of detecting sleep apnea, but it requires a smartphone and oximeter to be attached to the patient's body while sleeping. [18] successfully classifies the patients into those with apnea episodes and those without, with over 90% accuracy, but requires wearing an armband containing a phone, attaching a microphone on the face, and an oximeter to the wrist. Further, the classification scheme is limited to apnea / non-apnea subject classification, rather than detection of individual episodes of sleep apnea (medically defined 10-sec intervals) and types of apnea (central, obstructive and mixed).

Force sensors placed on top or under the mattress, have also been used for sleep monitoring and estimation of heart rate, respiration rate, snoring periods, etc (*e.g.*, [19]–[22]). There is no evidence, however, that such systems can differentiate among central, obstructive and mixed apnea.

The system in [23] estimates respiratory rate using received signals from wireless sensor nodes. However, the system requires a large number of wireless sensors to provide high accuracy (between 15 and 20 sensor nodes), only the test subject can be present in the room, and it is unclear if the system is accurate enough to detect apnea episodes based only on the detected breathing rate.

Video is used for non-contact sleep monitoring in [24]–[29]. Using video for sleep monitoring requires capturing the breathing action from the recorded images based on human pose estimation—a long-standing problem in computer vision [30]–[32]. For sleep monitoring, since color images are usually not available (due to the typically dark sleeping environment), and there is no clear separation between foreground object (patient under a blanket) and background (bed), colour-image hypergraph-distance [33] and pairwise-distance [34] based detection methods, and generic pose estimation techniques such as [31], [35], are not suitable for estimating the sleep pose.

[24], [25] adopt the camera-based optical approach using the MS Kinect infrared sensor. However, [24] is limited to respiration rate monitoring, and [25] is only

[3] We stress that we claim only to have observed correlation between relevant features extracted from recorded depth video and audio and respiratory events as scored by a scientific officer based on data collected by system Alice6 LdxS, and *not* actual medically defined sleep apnea symptoms or pathology.

validated on simulated respiratory events. The depth-video based sleep monitoring system of [26] is limited to sleep-awake status detection. [27] claims that a Time of Flight (ToF) camera was used to detect chest and abdomen movements for apnea detection, but there is no description of which ToF camera was used and how chest and abdomen movements were deduced from the collected depth measurements. There is also no performance analysis of the proposal against ground truth data. This renders a direct comparison with [27] impossible.

[28] describes in detail a sleep monitoring system that uses a single Kinect camera, where chest movements are detected by tracking over time the closest depth measurement of the patient to a virtual camera directly above the patient. We differ from [28] in three respects. First, we use both audio and video to infer respiratory events, which improves detection accuracy and enables us to distinguish among central, obstructive and mixed apnea. Second, we propose a complete system that includes efficient depth video coding and denoising schemes. Third, unlike [28], [29], [36]–[39], we propose a more accurate dual-ellipse model, so that individual chest and abdominal movements can be tracked, even if the patient is sleeping sideway.

B. *Comparison with Our Previous Work*

Compared to our previous proposed sleep monitoring systems [40], [41], we have the following non-trivial improvements. First, both video and audio features are extracted for sleep apnea detection, which improves detection accuracy (to be shown in Sec. VIII). Second, relevant video features are extracted using a WPT [8], [9], [42], which we show to outperform the hand-crafted features in [40], [41] (variances of major and minor axis of the dual-ellipse model) in event detection accuracy. Third, our new method based on the Bisection (BS) method [43]–[46] and Nelder-Mead simplex method [45], [47], [48] to determine parameters of the best fitting ellipse given observed depth pixel samples is much faster, and can now robustly track the patient's breathing patterns regardless of the sleeping pose (supine or sideway). With these improvements, we demonstrate that we can now differentiate between central apnea and obstructive / mixed apnea, which was not possible in our previous system.

III. System Overview

We first overview our proposed sleep monitoring system that employs an MS Kinect sensor to capture depth video and audio of a sleep patient. A potential usage of our system is as follows. When a patient stays overnight in a sleep clinic for initial testing, in addition to in-hospital system's sensors, we deploy also a Kinect sensor to capture depth video and audio for respiratory event classifier training. In subsequent nights at the patient's home, our proposed system that replicates the same Kinect sensor setup is activated to collect depth video and audio data non-intrusively for respiratory event classification. Without the body-attached sensors, this would mean a significant improvement in sleep comfort for the patient when at home.

Specifically, we employ a first-generation MS Kinect depth camera for depth video and audio processing and respiratory event classification. As shown in Fig. 1, the camera is set up at a higher elevation above and away from the head of the patient lying down. This camera location gives an unobstructed view of the patient's torso for depth video capture and analysis. The Kinect camera captures depth images of resolution 640 × 480 pixels with 11-bit pixel precision at 30 frames per second. The camera can also simultaneously capture audio at 16kHz, 16-bit sample precision with a PCM S16 LE audio codec [49]. Note that though Kinect camera has a 4-microphone array resulting in a 4-channel audio, we use only the first channel for recording.

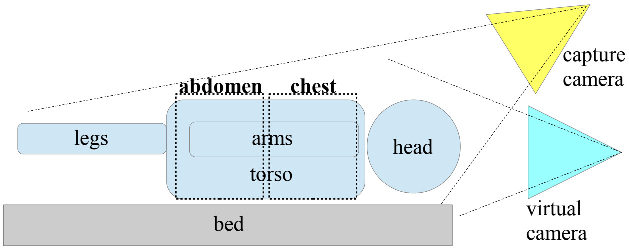


Fig. 1. Side view of sleep patient. Torso is divided into two cross sections, each modeled by an ellipse.

The first component of our system is the real-time capturing and compression of depth video (for transmission of captured video to a remote powerful server for storage and analysis) and recording of single-channel audio. We propose an efficient H.264 implementation of Kinect-captured video, where different 8 bits per pixel are extracted from 11 available bits of different temporal frames for encoding. At decoder, the uncoded 3 bits are recovered from neighboring frames via block motion search.

Second, we employ a graph-based temporal denoising algorithm to remove unwanted acquisition noise and flickers in recorded depth video. We show that the temporal flickers can be noticeably removed without over-smoothing and blurring of sharp edges typical in depth images.

Third, using the denoised depth video we track the chest and abdominal movements of the patient over time, as shown in Fig. 1. In a nutshell, we model the cross-sections of the patient's chest and abdomen as ellipses, and we derive ellipse parameters that best fit the observed depth pixels per frame. The changes of the ellipse parameters over time will reveal breathing cycles and patterns.

Finally, we perform WPT [8], [9] on the ellipse parameters to extract video features, and NMF [10]–[13] on the recorded audio to extract audio features. The extracted features are used to train an SVM classifier and a feed-forward NN with four event classes: i) central apnea, ii) obstructive or mixed apnea, iii) hypopnea, and iv) all the other events that are available from the ground truth labels. Fig. 2 illustrates the overall proposed system.

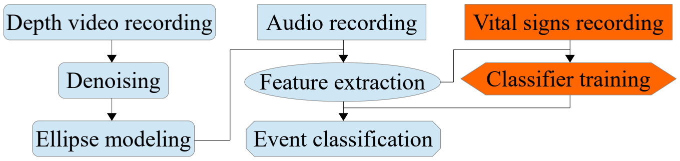
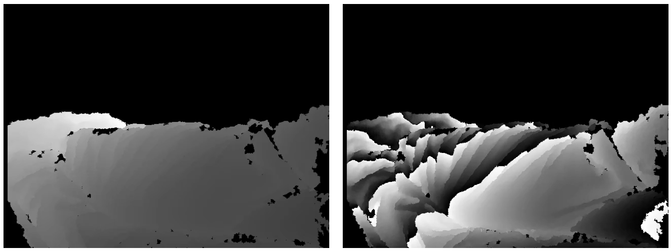


Fig. 2. System overview. 'Vital Signs Recording' is for ground truth; Orange: initial training components; Blue: regular usage components.

IV. Depth Video Recording

We now describe our proposed coding algorithm to compress captured depth videos of sleeping patients. Each depth image captured by a first-generation MS Kinect sensor contains 11-bit precision pixels at spatial resolution 640 × 480. *Baseline profile* for video coding standard H.264 [4]—the most prevalent and optimized profile—supports only 8-bit precision, however[4]. Thus, we propose an alternating frame coding scheme to extract different 8 of 11 available bits in each captured pixel of different frames for encoding. At the decoder, we recover the uncoded 3 bits using our proposed recovery scheme. The reasons we can recover the uncoded 3 bits with high accuracy are: i) depth maps are known to be *piecewise smooth* (PWS), and ii) in a typical sleep video, only slow motion exists across frames. We discuss the encoding and decoding procedures next.

A. Encoder Selection of 8 Coding Bits



(a) MSB frame (b) LSB frame

Fig. 3. Examples of MSB and LSB frames.

The encoder selects different 8 bits for each depth frame $\mathbf{Z}_t$ of time instant t for encoding as follows. Denote by M the *reference picture selection* (RPS) parameter used during H.264 video encoding [4]; *i.e.*, a P-frame $\mathbf{Z}_t$ can choose any one of the previous M frames $\mathbf{Z}_{t-1}, \ldots, \mathbf{Z}_{t-M}$ as predictor for differential coding. If $t \bmod M = 0$, then we select the 8 *most significant bits* (MSB) of 11 captured bits in each captured depth pixel in target frame $\mathbf{Z}_t$ for encoding. Otherwise, we select the 8 *least significant bits* (LSB) of 11 available bits in each pixel for encoding. MSB frames and LSB frames are very different; MSB frames are very smooth with missing details (contained in lost LSBs), while LSB frames suffer from overflow due to missing MSBs. See Fig. 3 for an illustration. However, our proposed encoding scheme ensures that each MSB or LSB frame $\mathbf{Z}_t$ can find a similar previous frame $\mathbf{Z}_{t-i}$ in predictor frame set $\{\mathbf{Z}_{t-1}, \ldots, \mathbf{Z}_{t-M}\}$ for differential coding thanks to RPS in H.264, thus achieving good coding efficiency (demonstrated in Sec. VIII-B1).

B. Decoder Recovery of Full 11 Bits

At the decoder, we recover the uncoded 3 MSBs in an LSB frame as follows. We first segment an LSB frame into *smooth regions*, *i.e.*, spatial regions where adjacent pixels do not differ by more than a pre-defined threshold δ. Pixels in the same smooth region will share the same to-be-recovered 3 MSBs.

Next, we identify potential *overflow* pixels in an LSB frame due to encoding of LSBs only—pixels that were similar to adjacent pixels before removal of 3 MSBs. Specifically, given smooth region boundary pixel location $\mathbf{p}$ in LSB frame $\mathbf{Z}_t$ where its pixel value is close to zero, *i.e.*, $\mathbf{Z}_t(\mathbf{p}) \leq \delta$, we check if adding one significant bit 2^8 would bring it closer to within δ of one of its neighbors, *i.e.*,:

$$\min_{\mathbf{q} \in \mathcal{N}_{\mathbf{p}}} \left| \mathbf{Z}_t(\mathbf{p}) + 2^8 - \mathbf{Z}_t(\mathbf{q}) \right|, \leq \delta \tag{1}$$

where $\mathcal{N}_{\mathbf{p}}$ is the set of adjacent pixels to $\mathbf{p}$. If this is the case, then $\mathbf{p}$ is a potential overflow pixel. To check if $\mathbf{p}$ is an overflow pixel (or simply an object boundary), we perform *motion estimation* (ME) [50] using the most recent MSB frame $\mathbf{Z}_\tau$. Specifically, given an $R \times R$ block $B_{\mathbf{p}}$ with center at $\mathbf{p}$ of the current frame $\mathbf{Z}_t$ as target, we compute:

$$\min_{\mathbf{v}} \left| \mathbf{Z}_\tau(B_{\mathbf{p+v}}) \bmod 2^5 - \left\lfloor \frac{\mathbf{Z}_t(B_{\mathbf{p}})}{2^3} \right\rfloor \right| + \mu |\mathbf{v}|, \tag{2}$$

where the 5 LSBs in block $B_{\mathbf{p+v}}$ of $\mathbf{Z}_\tau$ and the 5 MSBs in block $B_{\mathbf{p}}$ of $\mathbf{Z}_t$ are compared—only 5 bits are common between MSB and LSB frames. Note that we add the magnitude of the *motion vector* (MV) $\mathbf{v}$ as a regularization term, because for PWS images, there can be multiple vectors $\mathbf{v}$ with very small block differences. $|\mathbf{v}|$ means we favor the smallest motion block in frame $\mathbf{Z}_\tau$, which is reasonable due to low level of motion in sleep videos. μ is a parameter that trades off the block differential and the regularization terms.

Given the best MV $\mathbf{v}_{\mathbf{p}}$ computed in (2), we then check if $B_{\mathbf{p}+\mathbf{v}_{\mathbf{p}}}$ is smooth in $\mathbf{Z}_\tau$. If so, then pixel $\mathbf{p}$ in $\mathbf{Z}_t$ is deemed an overflow bit, and we merge the smooth region of $\mathbf{p}$ with the corresponding neighboring smooth region; *i.e.*, the merged smooth region will share the same MSBs. If not, then this is actually an object boundary, and we copy the 3 MSBs in $B_{\mathbf{p}+\mathbf{v}_{\mathbf{p}}}$ of $\mathbf{Z}_\tau$ to *all* pixels in the smooth region containing $\mathbf{p}$. Fig. 4 illustrates the above procedure of decoder recovery of full 11 bits.

[4] Only High 4:4:4 Profile, that leads to high encoding complexity, supports 11 to 14 bits precision.

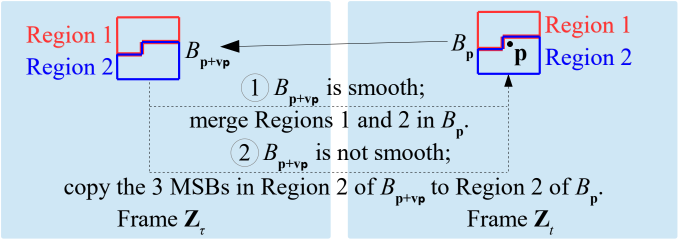

Region 1
Region 2
B_{p+vp}
B_p
p
Region 1
Region 2
① B_{p+vp} is smooth;
merge Regions 1 and 2 in B_p.
② B_{p+vp} is not smooth;
copy the 3 MSBs in Region 2 of B_{p+vp} to Region 2 of B_p.
Frame $\mathbf{Z}_\tau$
Frame $\mathbf{Z}_t$

Fig. 4. Decoder bits recovery given $\mathbf{p}$ is a potential overflow pixel.

V. Depth Video Denoising

Depth images captured by a Kinect camera are susceptible to acquisition noise and have missing pixel values especially around object boundaries, which can adversely affect the performance of subsequent sleep event classification. In this section, we propose a temporal denoising algorithm based on a graph-signal formulation. We show how a graph-signal smoothness prior can be used for temporal denoising in depth videos, which is more complex than spatial denoising [51] and involves the joint optimization of motion vectors (MV) and noise-corrupted pixels in the target frame.

We first formulate an optimization problem for the motion field in a frame t given previous frame $t-1$ and a motion smoothness prior. Then we discuss how the problem can be modified if frame t is corrupted by noise, and present an efficient algorithm to solve it.

A. Finding Motion Field

For simplicity, we assume first that neither target frame t nor previous frame $t-1$ is corrupted by noise. The goal is to find an accurate motion field for all $K \times K$ pixel blocks in frame t. Let $\mathcal{B}_{\mathbf{p}_i}(t)$ be the i-th $K \times K$ block in frame t, with upper-left pixel at $\mathbf{p}_i$. Let $\mathbf{v}_i = (x_i, y_i)$ be the MV of the i-th block. The MV field of all N blocks in the frame is expressed in vector form as $\mathbf{v} = [\mathbf{v}_1, \ldots, \mathbf{v}_N]$.

We first assume a *spatial motion smoothness prior*: a block's MV will be similar to MVs of neighboring blocks if they belong to the same object; *i.e.*, the MV field is PWS. One way of expressing piecewise smoothness is through a graph [5], [6], [52], [53]. We first construct a four-connected graph, where each node i represents a block $\mathcal{B}_{\mathbf{p}_i}(t)$ and is connected to nodes corresponding to neighboring blocks of $\mathcal{B}_{\mathbf{p}_i}(t)$. We compute the weight $w_{i,j}$ of an edge connecting two nodes (blocks) i and j as follows:

$$w_{i,j} = \exp\left\{-\frac{\|\mathbf{v}_i - \mathbf{v}_j\|_2^2}{\sigma_v^2}\right\}, \tag{3}$$

where σ_v is a chosen parameter. Given the constructed graph, we can define the *degree* and *adjacency* matrices, $\mathbf{D}$ and $\mathbf{A}$, correspondingly [7]. The *graph Laplacian* is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$. If the MV field is PWS, the *graph variation* term, $\|\mathbf{v}^\top \mathbf{L} \mathbf{v}\|_2^2$, is small: $\mathbf{v}^\top \mathbf{L}\, \mathbf{v} = \sum_{i,j} w_{i,j} \left(\mathbf{v}_i - \mathbf{v}_j\right)^2$. Note that because $\mathbf{v}_i$ contains x- and y-coordinates of the MV, $\|\mathbf{v}^\top \mathbf{L} \mathbf{v}\|_2^2$ means computing $\mathbf{v}^\top \mathbf{L} \mathbf{v}$ for the x- and y-coordinates, $\mathbf{v}(x)$ and $\mathbf{v}(y)$ of $\mathbf{v}$, separately, then computing the resulting vector magnitude square.

We can now define an optimal MV field as one that results in good block matches in the previous frame $t-1$ *and* is smooth with respect to the graph:

$$\min_{\mathbf{v}} \sum_i \|\mathcal{B}_{\mathbf{p}_i + \mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \;+\; \lambda\, \|\mathbf{v}^\top \mathbf{L} \mathbf{v}\|_2^2, \tag{4}$$

where λ is a chosen weighting parameter that trades off the ME term (first term) and the MV smoothness term (second term).

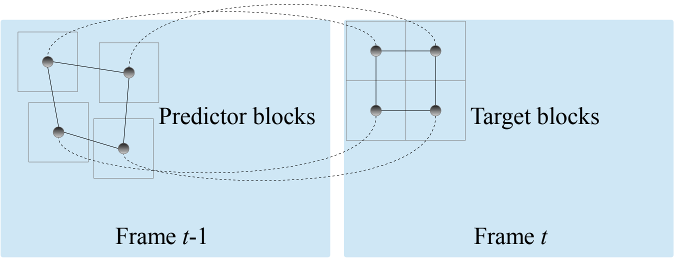


Fig. 5. Example graph construction given four blocks in target frame t and four corresponding predictor blocks in previous frame $t-1$.

B. Temporal Denoising

We now remove the earlier assumption that target frame t is noiseless, meaning we have to find MV field $\mathbf{v}$ *and* denoise blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ simultaneously. Beyond spatial MV smoothness prior, we now assume further a *temporal MV smoothness prior*; *i.e.*, if the i-th block at position $\mathbf{p}_i$ of frame t has MV $\mathbf{v}_i$, then the predictor block at position $\mathbf{p}_i + \mathbf{v}_i$ of frame $t-1$ will have a MV $\mathbf{u}_{\mathbf{p}_i + \mathbf{v}_i}$ that is similar to $\mathbf{v}_i$. We can again express this notion of smoothness via a graph. In particular, in addition to the graph constructed for MV $\mathbf{v}_i$ in frame t, we create additional nodes to represent predictor blocks in frame $t-1$. We draw an edge between node representing block $\mathcal{B}_{\mathbf{p}_i}(t)$ in frame t and node representing corresponding predictor block $\mathcal{B}_{\mathbf{p}_i + \mathbf{v}_i}(t-1)$ with weight computed by (3).

Furthermore, we draw an edge between two predictor blocks at locations $\mathbf{p}$ and $\mathbf{q}$ in frame $t-1$ if $\|\mathbf{p} - \mathbf{q}\|_2^2 \leq \Delta$, with edge weight computed as:

$$w_{i,j} = \exp\left\{-\frac{\|\mathbf{u}_\mathbf{p} - \mathbf{v}_\mathbf{q}\|_2^2}{\sigma_v^2}\right\} \exp\left\{-\frac{\|\mathbf{p} - \mathbf{q}\|_2^2}{\sigma_g^2}\right\}, \tag{5}$$

where σ_g is a chosen parameter. This weight assignment is similar to the one done in *bilateral filtering* [54]. See Fig. 5 for an example of a graph constructed from four blocks in the target frame t and four corresponding predictor blocks in the previous frame $t-1$.

Without loss of generality, we define the combined motion vector ζ to be a concatenation of MV $\mathbf{u}$ of predictor blocks of frame $t-1$ and MV $\mathbf{v}$ of target blocks of frame t, *i.e.*, $\zeta^\top = [\mathbf{u}^\top\ \mathbf{v}^\top]$. We can also define degree and adjacency matrices $\mathbf{D}$ and $\mathbf{A}$ as done previously for the larger graph. The resulting Laplacian $\mathbf{L}$ is again $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

With these definitions, we can define the new objective to find MV $\mathbf{v}$ and denoised blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ as a sum of three terms: i) ME error term, ii) MV smoothness term, and iii) fidelity term with respect to observed noisy blocks $\mathcal{B}^o_{\mathbf{p}_i}(t)$, *i.e.*,

$$\min_{\mathbf{v},\mathcal{B}(t)} \left\{ \begin{array}{l} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \;+\; \lambda \, \|\zeta^\top \mathbf{L} \zeta\|_2^2 \\ +\; \mu \sum_i \|\mathcal{B}_{\mathbf{p}_i}(t) - \mathcal{B}^o_{\mathbf{p}_i}(t)\|_2^2 \end{array} \right\}, \qquad (6)$$

where μ is a weighting parameter for the fidelity term. Note that, an ME error term (the first term in (6)) is introduced so that similar blocks can be identified between the previous and current frames. A regularization term (the second term in (6)) is employed to constrain the search space in an under-determined inverse problem. Finally, a fidelity term (the third term in (6)) is used to ensure that the denoised block is closed to the observation. We discuss how we solve (6) next.

C. Optimization Algorithm

(6) is difficult to solve as it involves many variables. Our strategy is to alternately solve one set of variables at a time while keeping the other set fixed, until convergence. Suppose first we initialize MV $\mathbf{v}$ using conventional ME [50], then fix $\mathbf{v}$ and solve for optimal blocks $\mathcal{B}_{\mathbf{p}_i}(t)$. The MV smoothness term is not affected by the selection of $\mathcal{B}_{\mathbf{p}_i}(t)$, and so (6) reduces to:

$$\min_{\mathcal{B}(t)} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 + \mu \sum_i \|\mathcal{B}_{\mathbf{p}_i}(t) - \mathcal{B}^o_{\mathbf{p}_i}(t)\|_2^2. \quad (7)$$

Let $\mathcal{B}_{\mathbf{p}_i}(t)$ be a convex combination of $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$ and $\mathcal{B}^o_{\mathbf{p}_i}(t)$, *i.e.*,

$$\mathcal{B}_{\mathbf{p}_i}(t) = \epsilon \, \mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1) + (1-\epsilon)\, \mathcal{B}^o_{\mathbf{p}_i}(t). \qquad (8)$$

By substituting (8) into (7), taking the derivative with respect to ϵ and setting the equation to zero, we see that the optimal ϵ^* is: $\epsilon^* = \frac{1}{1+\mu}$. This agrees with intuition; if $\mu = 0$, then $\epsilon^* = 1$ and $\mathcal{B}_{\mathbf{p}_i}(t)$ is set to predictor block $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$, and if $\mu = 1$, then $\epsilon^* = 1/2$, and $\mathcal{B}_{\mathbf{p}_i}(t)$ is the average of predictor block $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$ and observed noisy block $\mathcal{B}^o_{\mathbf{p}_i}(t)$.

Now we fix blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ and solve for the optimal MV $\mathbf{v}$. The fidelity term is not affected by MV $\mathbf{v}$, so (6) reduces to:

$$\min_{\mathbf{v}} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \;+\; \lambda \, \|\zeta^\top \mathbf{L} \zeta\|_2^2. \qquad (9)$$

(9) is still difficult to solve, since each change in MV $\mathbf{v}_i$ induces a change in corresponding predictor block $\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1)$, resulting in a different predictor MV $\mathbf{u}_{\mathbf{p}_i+\mathbf{v}_i}$ and a modified Laplacian $\mathbf{L}$. Our strategy then is to find first the optimal MV $\mathbf{v}^*$ that minimizes the smoothness term, then insert $\mathbf{v}^*_i$ into (9) to see if the objective is reduced.

Given ζ is a concatenation of predictor MV $\mathbf{u}$ and target MV $\mathbf{v}$, we can rewrite the smoothness term as:

$$\begin{aligned} & \underbrace{\begin{bmatrix} \mathbf{u}^\top & \mathbf{v}^\top \end{bmatrix}}_{\zeta^\top} \underbrace{\begin{bmatrix} \mathbf{L}_{\mathbf{uu}} & \mathbf{L}_{\mathbf{uv}} \\ \mathbf{L}_{\mathbf{vu}} & \mathbf{L}_{\mathbf{vv}} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}}_{\zeta} \\ = \;& \mathbf{u}^\top \mathbf{L}_{\mathbf{uu}} \mathbf{u} + \mathbf{u}^\top \mathbf{L}_{\mathbf{uv}} \mathbf{v} + \mathbf{v}^\top \mathbf{L}_{\mathbf{vu}} \mathbf{u} + \mathbf{v}^\top \mathbf{L}_{\mathbf{vv}} \mathbf{v}. \end{aligned} \qquad (10)$$

The first term is a constant and not influenced by $\mathbf{v}$. Additionally, $\mathbf{u}^\top \mathbf{L}_{\mathbf{uv}} \mathbf{v} = \mathbf{v}^\top \mathbf{L}_{\mathbf{vu}} \mathbf{u}$. Thus to find $\mathbf{v}^*$ that minimizes the smoothness term, we write: $\min_{\mathbf{v}} \mathbf{v}^\top \mathbf{L}_{\mathbf{vv}} \mathbf{v} + 2\mathbf{u}^\top \mathbf{L}_{\mathbf{uv}} \mathbf{v}$. This is an unconstrained quadratic programming problem, with closed form solution [55]:

$$\mathbf{v}^* = \mathbf{L}^{\#}_{\mathbf{vv}} \left(-\mathbf{u}^\top \mathbf{L}_{\mathbf{uv}}\right)^\top, \qquad (11)$$

where $\mathbf{L}^{\#}_{\mathbf{vv}}$ is the pseudo-inverse of $\mathbf{L}_{\mathbf{vv}}$.

Because $\mathbf{v}^*$ only minimizes the second term in objective (9), we perform the following greedy procedure using $\mathbf{v}^*$ to reduce the overall objective function value: we iteratively insert a maximally "beneficial" component of $\mathbf{v}^*$ (one that decreases the objective (9)) into the current vector $\mathbf{v}$. We stop when no more beneficial components in $\mathbf{v}^*$ exist.

Pixels in frame t, $\mathcal{B}(t)$, and MV $\mathbf{v}$ are alternately optimized using the two procedures described above, until the solution converges. Experimentation shows this only requires a few iterations in practice.

The proposed graph-based depth video temporal denoising scheme is summarized in Algorithm 1.

Algorithm 1 Graph-based depth video temporal denoising.

```
Input: Frames $t-1, t$;
Output: Denoised Frame $t$;
Initialise $\mathbf{u}$, $\mathbf{v}$;
while not converged do
    Optimise $\mathcal{B}_{\mathbf{p}_i}(t)$ in Frame $t$ by minimizing (7) given $\mathcal{B}^o_{\mathbf{p}_i}(t)$ and fixed $\mathbf{v}$;
    Optimise $\mathbf{v}$ by minimizing $\lambda \, \|\zeta^\top \mathbf{L} \zeta\|_2^2$ in (9) given $\mathbf{u}$ and optimised $\mathcal{B}_{\mathbf{p}_i}(t)$;
    Further optimise $\mathbf{v}$ by iteratively inserting maximally "beneficial" component of $\mathbf{v}^*$ (to minimize (9)) into current $\mathbf{v}$ until no more beneficial components in $\mathbf{v}^*$ exist;
end while
```

VI. Ellipse Modeling of Human Torso

In this section we discuss how we build our ellipse model in two steps using the denoised depth video. In the first step, each depth pixel from the captured camera view is mapped to a virtual camera view (*head-on view*) as illustrated in Fig. 1. To reduce the computation time, the region of interest is identified as a bounding box that contains only depth pixels of the patient, which is based on the difference of the depth images taken before and after the patient gets in bed. Each depth pixel with coordinate (u, v, d) in the virtual view is then classified into two different cross sections of the patient's torso—chest and abdomen—based on depth value d. See Appendix A for details of the above view transformation.

In the second step, we model each cross section (chest or abdomen) as an ellipse; *i.e.*, we estimate a best-fitting ellipse based on the set of observations (u, v)'s classified to this cross section. During regular breathing, the patient's chest and abdomen will expand and contract, resulting in ellipse size changes over time. We estimate the major and minor radii of ellipses per frame given observed depth video to track the patient's breathing cycle over time. Unlike our previous work [41], our new system can in addition detect the patient's body tilt during sleep (*e.g.*, sleeping on the side), resulting in rotated model ellipses about the origin. We describe how we formulate and solve the ellipse-fitting problem in detail next.

A. Problem Formulation

Let $\mathbf{o} = \{\mathbf{o}_1, \ldots, \mathbf{o}_N\}$ be the set of N observations for construction of one ellipse, where $\mathbf{o}_n$ is (u_n, v_n)—the observation's location in the u-v image coordinate system as observed from the virtual view. The parametrization of an ellipse in a Cartesian u-v coordinate system is:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} c_u \\ c_v \end{pmatrix} + \begin{pmatrix} \cos\delta & -\sin\delta \\ \sin\delta & \cos\delta \end{pmatrix} \begin{pmatrix} a \cos\phi \\ b \sin\phi \end{pmatrix}, \phi \in [0, 2\pi], \qquad (12)$$

where (c_u, c_v) denotes the center of the ellipse, a and b denote the major and minor radii, respectively, and δ denotes the ellipse tilt that models the patient's body tilt. For simplicity, we assume that the center of the ellipse is at the origin, *i.e.*, $c_u = c_v = 0$. An ellipse can thus be characterized by $\theta = (a, b, \delta)$.

Denote by $s_\theta(\mathbf{o}_n)$ the *minimum Euclidean distance* between observation $\mathbf{o}_n$'s location (u_n, v_n) and the ellipse with parameter θ. We formulate the following objective to find the best-fit ellipse parameters θ^* given observations $\mathbf{o}$:

$$\theta^* = \arg\min_\theta \sum_{n=1}^{N} s_\theta^2(\mathbf{o}_n). \qquad (13)$$

For example, for an ellipse with $\theta = (a, b, \delta)$, $s_\theta(\mathbf{o}_n) = \|(u_n, v_n) - (u_{\min}, v_{\min})\|_2$, where $(u_{\min}, v_{\min})$ is the closest point on the ellipse to (u_n, v_n); *i.e.*, the vector (u_n, v_n) to point $(u_{\min}, v_{\min})$ on the ellipse is orthogonal to the tangent of the ellipse at $(u_{\min}, v_{\min})$ [44], [56]. See Fig. 6 for an illustration of an ellipse with $\theta = (a, b)$.

B. Optimization Algorithm

Conventionally, (13) can be computed via either geometric ellipse fitting in parametric form by solving an equivalent nonlinear least squares problem, or fast algebraic ellipse fitting with geometric distance weighting [57]. Neither of these two approaches require initial ellipse parameters θ. However, the former can be very inefficient when building Jacobian due to large number of $\mathbf{o}_n$'s, and the latter does not generally minimize the geometric distance.

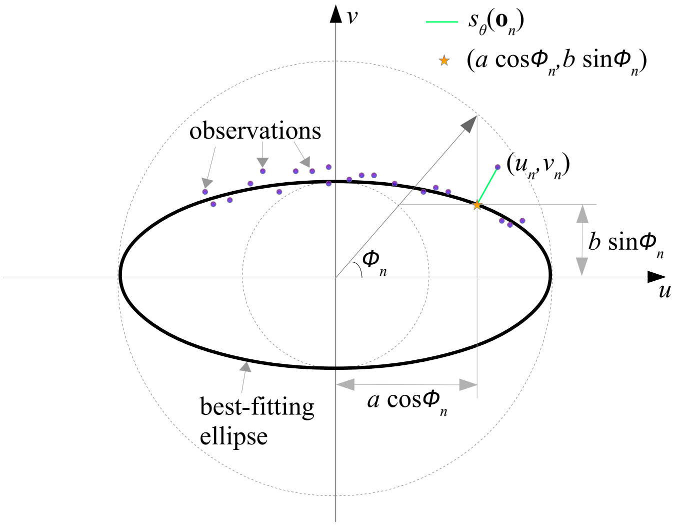


Fig. 6. Best-fitting ellipse from multiple depth observations of the cross section. The closest ellipse point to each observation is perpendicular to the tangent of ellipse at that point.

Note that, computing each $s_\theta(\mathbf{o}_n)$ given initial ellipse parameters θ is a well-known *root-finding* problem, which can be solved by solving a quartic equation with four roots [56], [58]. The root that is closest to (u_n, v_n) is then chosen to determine ϕ_n. However, this is clearly inefficient given a large number of $\mathbf{o}_n$'s. Instead, we adopt the Bisection (BS) method [43]–[46]. (See Appendix B.) We choose the BS method instead of Newton's Method as done in [40], [41], because the latter has numerical problems when v_n is nearly zero.

Since $\sum_{n=1}^{N} s_\theta^2(\mathbf{o}_n)$ is non-convex, we resort to a local numerical method—the Nelder-Mead (NM) simplex method [45], [47], [48]—to find the best ellipse parameters θ^* in (13), using $s_\theta(\mathbf{o}_n)$ found by the BS method explained above. See Appendix C for details.

VII. Feature Extraction & Classification

In this section, we describe how to extract relevant features from the depth video signal (*i.e.*, the four computed 1D signals—major and minor radii of the two fitted ellipses (chest and abdomen) as functions of time) and audio signal. We note that the time duration for each experimental data segment for feature extraction—both the computed 1D ellipse signal segments $\mathbf{x}$ and the audio signal segment $\mathbf{y}$—is set at 10 sec, which is the medically defined duration of a respiratory event [59]. The segment window is then shifted by 5 sec, so neighboring segments have a 5-sec overlap.

A. Depth Video Features

Unlike our previous work [40], [41] where we directly used the variances of the ellipses' major and minor radii in a time window to perform classification, in this paper, we adopt wavelet analysis, namely, WPT [8], [9], [42].

WPT decomposition adopts recursive splitting of vector spaces that is represented in a binary tree (see Fig. 8.1 in [9] for an example), which produces a redundant representation by using analysis filters for both high and low frequencies [60], [61].

In particular, each sub-segment $\boldsymbol{a} \in \mathbb{R}$ (with the size defined in Sec. VIII-B3) of the 10-sec 1-D ellipse signal segment $\mathbf{x}$ (*i.e.*, the amplitude of the ellipse major/minor radius over time), is approximated at the scale 2^J, *i.e.*, at J levels (where $J \in [0, \log_2 N]$, with N being the number of samples in $\boldsymbol{a}$). Each level j contains N approximation and detail coefficients that are divided into 2^j tree-nodes, and each tree-node thus contains $N/2^j$ coefficients.

After this WPT signal decomposition, we concatenate the normalized logarithmic energy [42] of each coefficient in the increasing order of the tree-nodes resulting in the feature vector $\tilde{\mathbf{E}}_i$ for the i-th sub-segment $\boldsymbol{a}_i$ of the original 1D ellipse signal segment $\mathbf{x}$. Finally, we concatenate all $\tilde{\mathbf{E}}_i$ forming a feature vector $\tilde{\mathbf{E}} = \left[\tilde{\mathbf{E}}_1, \ldots, \tilde{\mathbf{E}}_P\right]$ for $\mathbf{x}$, where P is the number of sub-segments $\boldsymbol{a}_i$ of $\mathbf{x}$.

B. Audio Features

For audio feature extraction we resort to NMF [10], [12], which is commonly used for audio feature extraction. Indeed, NMF is frequently used in spectral data analysis [13], and audio signals are well-fit for such feature extraction method given their spectrograms. By the virtue of nonnegativity [12], NMF is able to unsupervisedly learn parts representation of the signal, in contrast to other methods, such as Principle Component Analysis (PCA) and vector quantization, that learn holistic, distributed representations [11].

We perform NMF decomposition in the following way. We first apply short-time Fourier transform (STFT) on each sub-segment $\boldsymbol{b}$ (with the size defined in Sec. VIII-B3) of the 10-sec 1-D audio signal segment $\mathbf{y}$, resulting in a spectrogram matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$ as the magnitude of STFT. Then, we solve the NMF problem, *i.e.*, find a spectral-feature matrix $\mathbf{W} \in \mathbb{R}^{m \times k}_{\geq 0}$ and a temporal-activity matrix $\mathbf{H} \in \mathbb{R}^{k \times n}_{\geq 0}$ by minimizing the following cost function:

$$D(\mathbf{B}|\ \mathbf{WH}) = ||\mathbf{B} - \mathbf{WH}||^2, \tag{14}$$

where the product $\mathbf{WH}$ is an approximate factorization of $\mathbf{B}$ at rank k. We discuss how to choose an appropriate rank k in Sec. VIII-B3.

An alternating least-square (ALS) update rule is used to find the optimal matrices $\mathbf{W}$ and $\mathbf{H}$. Specifically, following [13], we initialize $\mathbf{W}$ as an $m \times k$ random dense matrix, then iteratively solve for $\mathbf{H}$ based on $\mathbf{W}^\top \mathbf{WH} = \mathbf{W}^\top \mathbf{B}$, followed by a projection step, *i.e.*, setting all negative elements within $\mathbf{H}$ to 0. Next, we solve for $\mathbf{W}$ based on $\mathbf{HH}^\top \mathbf{W}^\top = \mathbf{HB}^\top$, followed by the same projection step on $\mathbf{W}$. The above ALS rule with projection steps aids sparsity, converges faster and performs more consistently comparing with multiplicative update rules [13]. To alleviate the uniqueness problem which can be easily seen by considering $\mathbf{WDD}^{-1}\mathbf{H}$ for any non-negative nonsingular matrix $\mathbf{D}$ [13], given $\mathbf{W}$ and $\mathbf{H}$ after each iteration, we first normalize them as $\hat{\mathbf{W}} = \mathbf{WD}$ and $\hat{\mathbf{H}} = \mathbf{D}^{-1}\mathbf{H}$, respectively, where $\mathbf{D} = \text{diag}(\sqrt{\sum_{u=1}^{n} \mathbf{H}(1,u)^2}, ..., \sqrt{\sum_{u=1}^{n} \mathbf{H}(k,u)^2})$. Then, for obtaining a consistent permutation, we reorder the columns of $\hat{\mathbf{W}}$ as $\tilde{\mathbf{W}}$ by the index of the decreasing magnitude of the elements in $\dot{\mathbf{W}} = \left[\sum_{u=1}^{m} \hat{\mathbf{W}}(u,1)^2, ..., \sum_{u=1}^{m} \hat{\mathbf{W}}(u,k)^2\right]$, followed by reordering the rows of $\hat{\mathbf{H}}$ as $\tilde{\mathbf{H}}$ accordingly.

We perform NMF decomposition on $\boldsymbol{b}_i$ using the designated rank k, reshape $\tilde{\mathbf{W}}$ as $\breve{\mathbf{W}} = \left[\tilde{\mathbf{W}}(:,1)^\top, \ldots, \tilde{\mathbf{W}}(:,k)^\top\right]$, reshape $\tilde{\mathbf{H}}$ as $\breve{\mathbf{H}} = \left[\tilde{\mathbf{H}}(1,:), \ldots, \tilde{\mathbf{H}}(k,:)\right]$, concatenate $\breve{\mathbf{W}}$ and $\breve{\mathbf{H}}$ as the feature vector $\tilde{\mathbf{U}}_i = [\breve{\mathbf{W}}, \breve{\mathbf{H}}]$ for the i-th sub-segment $\boldsymbol{b}_i$ of the original 1-D audio signal segment $\mathbf{y}$. Finally, we concatenate all $\tilde{\mathbf{U}}_i$ forming a feature vector $\tilde{\mathbf{U}} = \left[\tilde{\mathbf{U}}_1, \ldots, \tilde{\mathbf{U}}_Q\right]$ for $\mathbf{y}$, where Q is the number of sub-segments $\boldsymbol{b}_i$ in $\mathbf{y}$.

C. Classification

Next, we train classifiers using $\tilde{\mathbf{E}}$ and $\tilde{\mathbf{U}}$, our extracted relevant depth video and audio features, respectively, for respiratory event classification. We train an SVM with a linear kernel, since given $\tilde{\mathbf{E}}, \tilde{\mathbf{U}} \in \mathbb{R}^{z \times 1}, z > 2000$, *i.e.*, the number of features is large, it is preferable to use linear kernel, *i.e.*, mapping data to a higher dimensional space does not improve the performance (see Appendix C in [62]). Since SVM does not include a feature selection process, we also train a feed-forward NN with sigmoid hidden neurons and softmax output neurons, to investigate if training a classifier that involves nested subset feature selection methods can improve classification performance and cost-effectiveness [63], [64] for our high-dimensional datasets. We present our classification results in Sec. VIII-B3.

VIII. Experimentation

A. Experimental Configurations

We captured a 480-minute depth video and audio for each patient with suspected sleep apnea at Concord Private Hospital in Sydney, Australia during January and February 2015[5]. The data were collected from four consenting patients over a two-day period. The data used for training and testing SVM and NN classifiers is limited to sleep periods (including wake periods that occurred during sleep periods)—382 ± 37 minutes for each subject. Besides our depth video and audio capturing, each patient was connected to the Alice6 LDxS as used in the corresponding attended diagnostic sleep studies. The sleep studies were attended polysomnography, and the scientific officer who scored the sleep studies was blinded to our multimedia feature learning study. The data obtained from the system was manually scored according to the AASM 2007 manual [3] and the

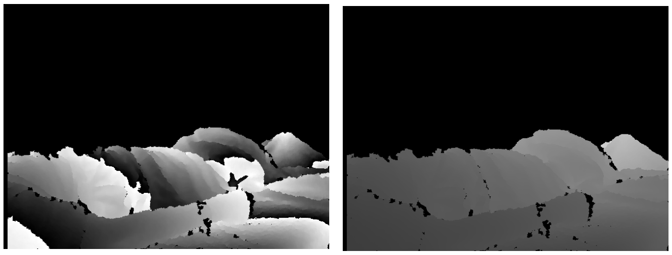
[5] The experimental procedure performed using depth video and audio has passed the ethical committee in National Institute of Informatics (NII) in Tokyo, Japan.

respiratory events were identified. These event labels are the ground truth data for our experiments. For a respiratory event that is of over 10-second length, we used the same segment window (10-second in length with a 5-sec overlap) as we used in the video and audio data to get data segments that have the same class as that event.

We present experimental results in the following order: depth video compression, depth video denoising, and respiratory event detection.

B. Experimental Results

1) Depth Video Recording: We first validate our proposed block-based search procedure to recover the 3 uncoded MSBs in an LSB frame. We set block size to 8 × 8 (see Sec. IV). Fig. 7 shows an example of the decoded LSB frame and the recovered LSB frame. First, we see in Fig. 7(a) that due to overflows, there are discontinuities even within the same physical object. We see in the recovered LSB frame in Fig. 7(b) that the overflow problem is corrected, resulting in a much smoother and natural looking depth image.



(a) original LSB frame (b) recovered LSB frame

Fig. 7. Examples of decoded LSB frame and recovered LSB frame.

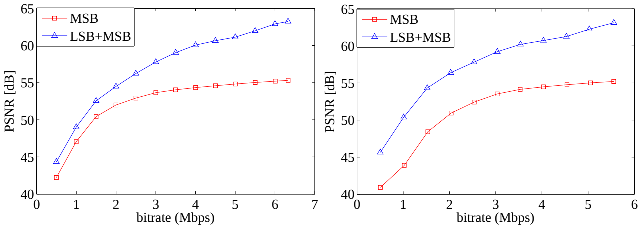
Next, we compare compression performance of our LSB-MSB coding scheme with RPS parameter $M = 5$ to the scheme that compresses only the 8 MSBs of each depth frame using the same H.264 implementation—AVC part 10 codec [4]. As a performance metric we used PSNR, calculated as:

$$\mathrm{PSNR} = 10 \log_{10} \frac{(2^{11}-1)^2 \cdot X \cdot Y}{\sum_{i=1}^{X}\sum_{j=1}^{Y}[\mathcal{X}(i,j) - \mathcal{Y}(i,j)]^2}, \qquad (15)$$

where $\mathcal{X}$ and $\mathcal{Y}$ are two $X \times Y$ pixel 11-bit depth images. Uncompressed 11-bit depth images were used as ground truth, and for the 8-MSB coding scheme, three zero bits were appended to the decompressed 8-bit values.

Fig. 8 shows the coding performance as PSNR averaged over all frames of the two coding schemes for two sleep video sequences. The results indicate that our LSB-MSB coding scheme outperforms 8-MSB coding scheme for up to 8dB.

2) Depth Video Denoising: We next evaluate the performance of our proposed graph-based temporal denoising scheme in terms of flickering reduction. Table I lists the parameter settings for our denoising scheme (see Sec. V). For comparison, we used the following as competing schemes. The first scheme is bilateral filtering (BF) [54] that performs spatial filtering using local neighboring pixels. We also implemented an algorithm that performs motion estimation and temporal median denoising (TM-F) separately, similar to existing works such as [65]. Additionally, we performed weighted mode filtering (WMF) [66] and tested an augmented Lagrangian-based (AL) video denoising algorithm [67].



(a) Video sequence 1. (b) Video sequence 2.

Fig. 8. Compression performance for two sleep video sequences.

TABLE I

PARAMETER SETTINGS OF THE PROPOSED GRAPH-BASED TEMPORAL VIDEO DENOISING SCHEME.

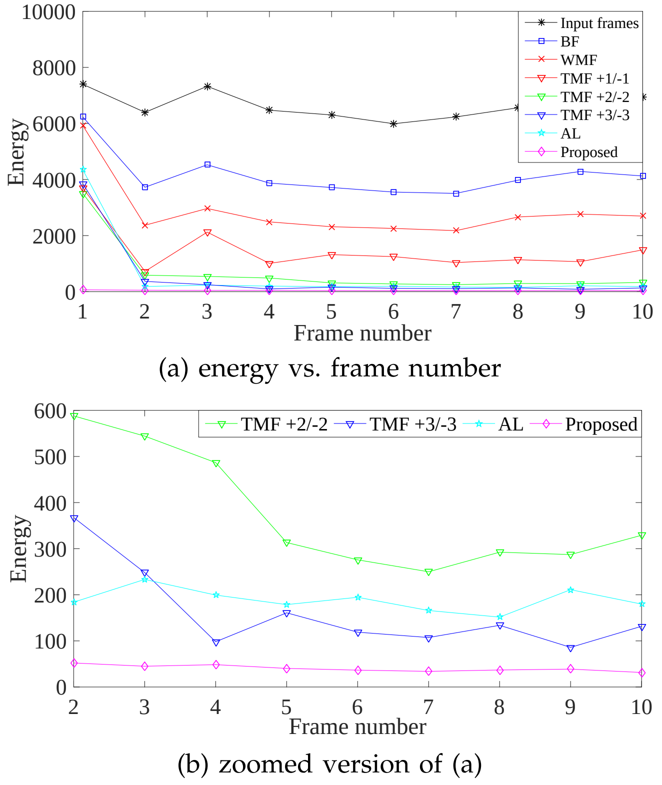
sign	parameter	setting
S	block size in pixels	8
Δ	thresholding for predictor-block edge	5
σ_v	target-block edge weight scaling	1
σ_g	predictor-block edge weight scaling	1
μ	weight for the fidelity term	0.1
λ	weight for the MV smoothness term	1

Fig. 9 shows the energy of the difference between two consecutive frames for our scheme and the competing schemes for the first 10 frames of an acquired sleep video sequence. We observe that our scheme is lowest in frame-difference energy for each of the tested consecutive frames.

Fig. 10 shows an example of a zoomed segment of a denoised depth frame using AL [67] and our proposed denoising scheme. We observe that our scheme preserves sharp edges without over-smoothing.

3) Respiratory Event Detection: We first performed four-class classification—i) central apnea, ii) obstructive / mixed apnea, iii) hypopnea, and iv) all the other events—using depth video features extracted from the 1-D signals based on our dual-ellipse model. For each 10-sec segment **x**, we used a sub-segment size of 5-sec with 0.5-sec increments and performed WPT at $J = 5$ levels on each sub-segment. To train a four-class SVM classifier, we adopted one-against-one strategy by training six binary SVM classifiers, a competitive approach among five multi-class SVM classification methods compared in [68]. We trained a two-layer feed-forward NN with 10 sigmoid hidden neurons and 4 softmax output neurons as a competing classifier. We also used the variance (VAR) of the ellipse major/minor radius as depth video features [40], [41] for training the same classifiers.

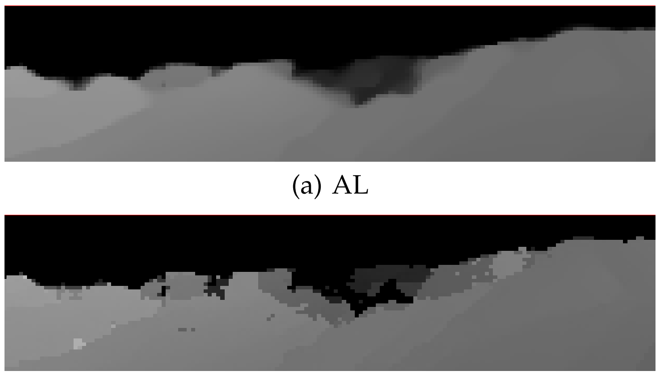
Fig. 11 shows the classification error rates of inverse



(a) energy vs. frame number

(b) zoomed version of (a)

Fig. 9. Energy of the difference between two consecutive frames, where +*i*/-*i* denotes the number of future and previous depth images used for TMF.



(a) AL

(b) proposed

Fig. 10. Sample segments of denoised frames by using AL and proposed scheme.

5-fold cross-validation (CV) (each time using 1-fold for training and the remaining 4-folds for testing), inverse 3-fold CV, 3-fold CV, and 5-fold CV based on video features only. We see that the classifiers with WPT features significantly outperform the hand-crafted VAR features in [40], [41]. Fig. 12 demonstrates a 300-minute sample of a sleep patient showing the major/minor radius and the tilt of the chest/abdomen ellipse, with groundtruth-sleeping-poses marked side-by-side. One can see that our system can robustly track the patient's respiratory patterns regardless of the sleeping pose, and $\delta_{abdomen}$ shows strong correlation with the actual sleeping pose. Fig. 13 shows the successfully detected respiratory events using WPT depth video features during the sideway sleep period that is highlighted in Fig. 12. In particular, the colourised bars at the top of the figures denote the manually scored events by a scientific officer based on data collected by system Alice6 LDxS[6]; the plotted lines denote the major and minor radii of the fitted ellipses for the patient's chest and abdominal cross sections, and the colours on the plotted lines are the detected respiratory events by our learned classifier.

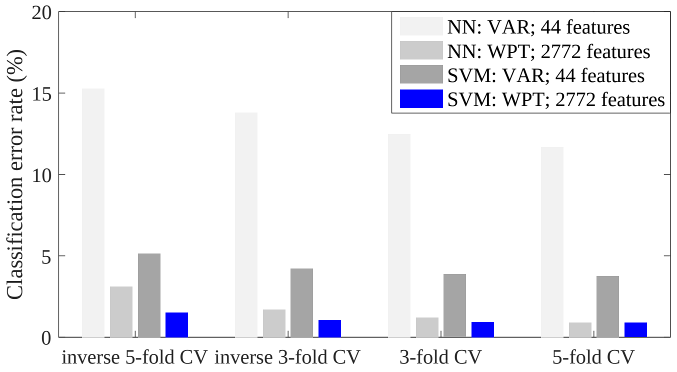


Fig. 11. Error rates of classification based on depth video features.

For four-class respiratory event classification with audio features, we heuristically set the rank $k = 3$ for NMF feature extraction (see our discussion in Section VIII-C2). As competing feature sets we use the following two sets: i) We apply WPT at $J = 7$ levels (each 10-sec 1-D audio signal segment $\mathbf{y}$ has much more elements than $\mathbf{x}$) on each segment of $\mathbf{y}$'s and training classifiers since such biomedical audio signals also contain different types of time-frequency structures [9]. ii) We concatenate the following conventional audio features as a MIX audio feature vector and train classifiers, namely, energy, energy entropy, harmonic ratio, fundamental frequency, spectral centroid, spectral entropy, spectral rolloff, spectral flux, zero crossing rate, Mel-frequency cepstral coefficients and chroma vectors [69]. Fig. 14 shows the classification error rates based on audio features only. Both SVM and NN classifiers trained by using NMF features show their best performance.

Finally, we train SVM and NN classifiers by combining both depth video and audio features used above. One can see in Fig. 15 that both classifiers perform better than using the features extracted from either of the two media, where the combination WPT+NMF shows

[6] In our experiments, an apneic event containing periods which fulfill hypopnea rules and do not fulfill apnea rules are treated as multiple individual events, e.g., we treat an apneic event that begins with a period that fulfills hypopnea rules followed by an immediate following period that fulfill apnea rules as two individual events - a hypopnea event with an immediate following apnea event. After the individual respiratory events are correctly classified, it is straightforward to automatically combine a hypopnea event with an immediate following apnea into a single apnea event, as specified in AASM recommendations [3].

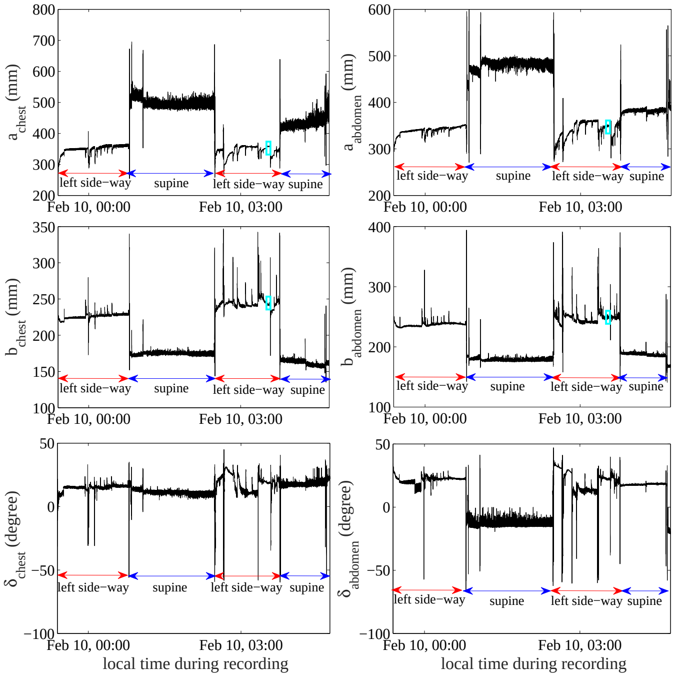


Fig. 12. 300-minute sample of a sleep patient showing six ellipse parameters over time. a_{chest} and b_{chest} are the major and minor radii of the chest-ellipse, respectively; $a_{abdomen}$ and $b_{abdomen}$ are the major and minor radii of the abdomen-ellipse, respectively; $\delta_{abdomen}$ and δ_{chest} are the tilts of the abdomen-ellipse and the chest-ellipse, respectively.

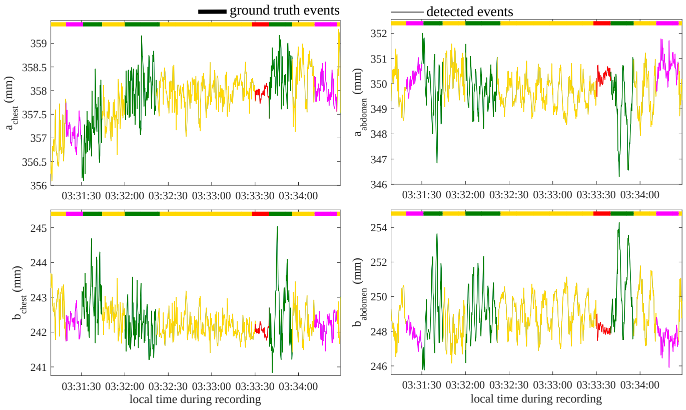


Fig. 13. Successfully detected events based on WPT depth video features showing $a_{chest}, b_{chest}, a_{abdomen}$ and $b_{abdomen}$ during the sideway sleep period that is highlighted in Fig. 12. Red: central apnea; Magenta: obstructive and mixed apnea; Yellow: hypopnea; Green: other events.

the best performance with inverse 5-fold CV error rates of only 0.4% and 1.67%, for SVM and NN classifiers, respectively. Table II shows the inverse 5-fold CV error rates of SVM classification based on the above three sets of features: WPT depth video feature, NMF audio feature, and WPT video+NMF audio feature.

Additionally, for each class we compute *sensitivity* and *specificity* of SVM classification based on WPT video+NMF audio feature with cross-validation, which are defined as follows:

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}}, \qquad (16)$$

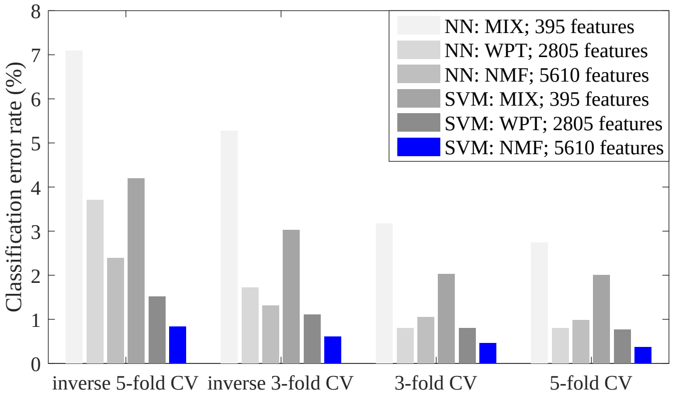


Fig. 14. Error rates of classification based on audio features.

where true positive (TP) denotes that a central apnea (resp. obstructive or mixed apnea, hypopnea and all the other events) testing sample is correctly classified, false positive (FP) denotes that a non-central apnea testing sample is incorrectly classified as central apnea, true negative (TN) denotes that a non-central apnea testing sample is correctly classified as non-central apnea, and false negative (FN) denotes that a central apnea testing sample is incorrectly classified as non-central apnea. The results are shown in Figs. 16 and 17 based on inverse 5-fold CV, inverse 3-fold CV, 3-fold CV, and 5-fold CV, respectively. The minimum sensitivity of the trained classifier is 98.2% for central apnea in inverse 5-fold CV and minimum specificity 99.76% for all the other events in 3-fold CV.

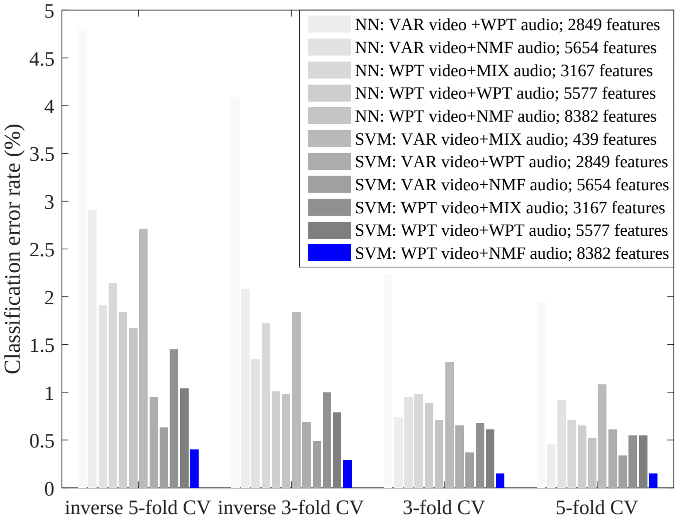


Fig. 15. Error rates of classification based on depth video+audio features.

TABLE II
INVERSE 5-FOLD CV ERROR RATES OF SVM CLASSIFICATION BASED ON WPT VIDEO FEATURES, NMF AUDIO FEATURES, AND THE COMBINATION OF THEM.

features	video	audio	video + audio
error rates	1.52%	0.83%	**0.4%**

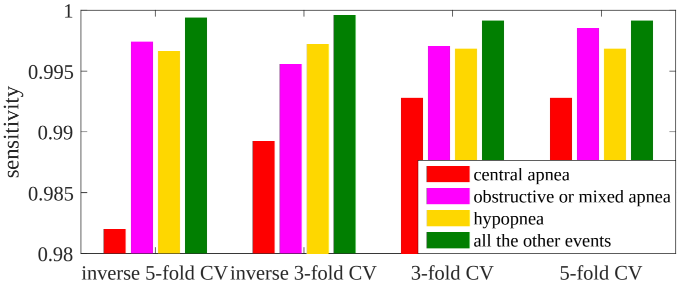


Fig. 16. The sensitivity of classifying different respiratory events using a trained SVM classifier based on WPT video+NMF audio feature with CV.

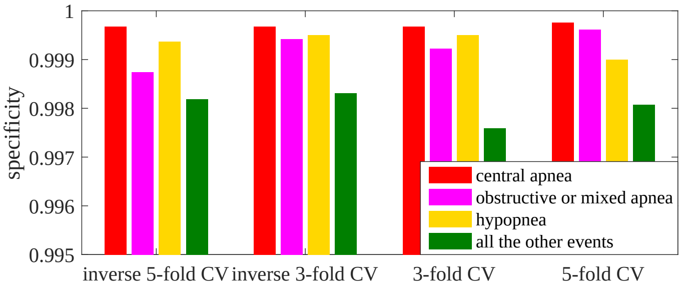


Fig. 17. The specificity of cross-validation classifying different respiratory events using a trained SVM classifier based on WPT video+NMF audio feature with CV.

C. Discussions

1) Depth Video Recording and Denoising: First, our LSB-MSB depth video compression scheme outperforms 8-MSB coding scheme at the PSNR range of sufficient quality for respiratory event detection. Second, our graph-based temporal denoising scheme can more effectively reduce frame-difference energy, and thus flickering effects, over the competing schemes, even if fewer number of frames were used in the processing window than competing schemes; while our denoising scheme reduces the flickering effect, it does not over-smooth and preserves sharp edges well.

2) Respiratory Event Detection: For respiratory event detection with video features, we compared the performance of our previous Newton's Method-based ellipse-fitting scheme [40], [41] and the proposed Bisection method and Nelder-Mead simplex method-based (BSN-M) scheme (Sec. VI-B) in terms of the computation speed. We ran both algorithms on 100 consecutive depth video frames in MATLAB R2014b on a Windows 10 laptop with Intel Core i7-4600U and 8GB RAM, and report that the average computation time per ellipse is 36.53s using [40], [41] and 8.68s using BSNM, *i.e.*, there is a 76% speed-up and also one can get ellipse-tilts in addition to major/minor radius, by using the new BSNM ellipse-fitting method.

Next, we built a competing dual-rectangle model and compared it to our dual-ellipse model in terms of the classification performance. Specifically, given observations $\mathbf{o}$, we found the best-fit rectangle ϱ^*, $\varrho = (\alpha, \beta, \varpi)$, with α, β and ϖ denoting the length, width and the tilt that represents the body pose, using the objective that is similar to (13):

$$\varrho^{best} = \arg\min_{\varrho} \sum_{n=1}^{N} h_n^{-1}\left(s_{\varrho}(\mathbf{o}_n)\right)^2 . \qquad (17)$$

We used the same video clips as in [40], [41], and trained SVM classifiers using similar hand-crafted features, *i.e.*, the variances of four ellipse major/minor radius for dual-ellipse model and those of four rectangle length/width for dual-rectangle model, for fair comparison. We performed binary classification (*i.e.*, Class 1: central / obstructive / mixed apnea / hypopnea; and Class 2: all the other events) with 50% data used for training and the remaining 50% for testing. The resulting confusion matrices (in the following format: [true positive, false positive; false negative, true negative]), $[50\%, 0\%; 0\%, 50\%]$ and $[10\%, 6\%; 40\%, 44\%]$, for the dual-ellipse and dual-rectangle model, respectively, show significant performance advantage of using our dual-ellipse model.

For respiratory event detection with audio features, we justify how we set the rank k for NMF feature extraction. For each 10-sec segment $\mathbf{y}$, we used the same sub-segment size (5-sec with 0.5-sec increments). For the i-th 5-sec sub-segment $\boldsymbol{b}_i$, we computed its spectrogram $\mathbf{B}$ by STFT with 25ms STFT-window and 12.5ms increments. We first applied singular value decomposition (SVD) on all $\mathbf{B}$'s. Fig. 18 shows the mean singular values of all $\mathbf{B}$'s. One can see that the majority of the singular values are small.

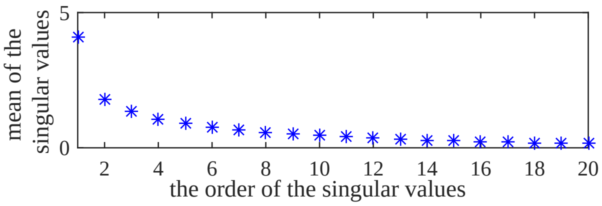


Fig. 18. The first 20 of the mean singular values of all $\mathbf{B}$'s.

TABLE III
5-FOLD CV ERROR RATES OF SVM CLASSIFICATION BASED ON NMF AUDIO FEATURES WITH $k = 2, 3, 4$.

sample handling	features	$k = 2$	$k = 3$	$k = 4$
with sub-segments	5610	0.46%	**0.34%**	0.4%
no sub-segments	630	0.77%	**0.65%**	1.01%

Since there is no clear dropoff between these singular values, in Table III, we present the 5-fold CV error rates of SVM classification based on NMF audio features using k = 2, 3 and 4. Specifically, we extracted NMF features from $\mathbf{B}$'s, trained SVM classifiers, and show the classification error rates in the row 'with sub-segments' in Table III; we also extracted NMF features from the spectrograms that were generated by performing STFT on each complete 10-sec $\mathbf{y}$'s and trained SVM classifiers, with the classification error rates shown in the row 'no sub-segments'. Given the fact that classifier always performs best at $k = 3$, we set $k = 3$ for our subsequent classification experiments. This is consistent with our initial hypothesis that the audio contains: i) background noise, ii) machine sound (*e.g.*, the cooling module of the system), and iii) human sound.

The trained classifiers with WPT video features outperforms the hand-crafted VAR features in our prior work. The classification with NMF audio features indicates that when the captured depth video is obstructed, one can still use the audio signal to detect respiratory events. Finally, the result of sensitivity and specificity for SVM classification with video-audio features reported in Figs. 16 and 17 indicates that our trained classifier has good ability to both correctly identify a central apnea (resp. obstructive or mixed apnea, hypopnea and all the other events) and correctly identify a non-central apnea, with 20% or more training data.

IX. Conclusion

Existing sleep monitoring systems are expensive and intrusive enough that they negatively affect the quality of a patient's sleep. In this paper, we propose to record audio and depth video of a patient using a Microsoft Kinect camera during his/her sleep, so that relevant features can be extracted non-intrusively for detection of different respiratory events. Our proposal contains three parts. First, we propose an efficient H.264 video coding scheme, where the captured 11-bit video can be reliably recovered at the decoder even though the compressed video is first converted to 8-bit. Second, we propose a graph-based depth video denoising algorithm, so that undesirable flicker can be removed without over-smoothing. Third, we propose a dual ellipse model to track the patient's chest and abdominal movements given captured depth pixels. When ellipse features are combined with audio features, different respiratory events, as scored manually based in data collected by system Alice6 LdxS, can reliably be detected.

We note that, our system requires large storage for data recording, it is relatively slow in fitting of the dual-ellipse respiratory model and person-specific classifier training for each human subject. Using large amount of collected data, future work would focus on developing more efficient and less complex model fitting methods and feature extraction for training classifiers that are generally applicable to different subjects.

Appendix A
View transformation

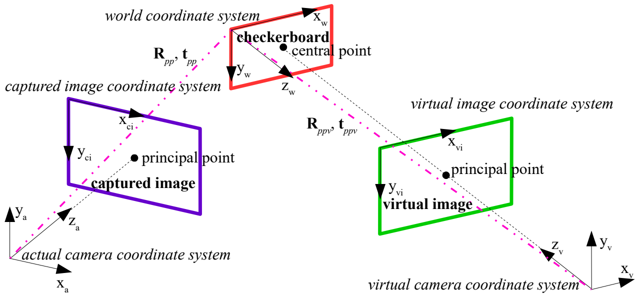

world coordinate system
checkerboard
central point
x_w
y_w
z_w
$\mathbf{R}_{pp}$, $\mathbf{t}_{pp}$
captured image coordinate system
x_{ci}
y_{ci}
principal point
captured image
y_a
z_a
x_a
actual camera coordinate system
$\mathbf{R}_{ppv}$, $\mathbf{t}_{ppv}$
virtual image coordinate system
x_{vi}
y_{vi}
principal point
virtual image
z_v
y_v
x_v
virtual camera coordinate system

Fig. 19. View transformation setup.

As shown in Fig. 19, we follow [70], set the origin of the *world coordinate system* to the upper-left feature point of a checkerboard, fix the actual camera - Kinect, and capture n infrared images and n corresponding depth images with different checkerboard orientations, including a pair of infrared and depth images showing that the checkerboard is closely perpendicular (pp) to the centerline of the virtual view, l mm away from the virtual camera, denoted as I_{pp} and D_{pp}, respectively. Each infrared image, denoted as I_i, is formed by projecting pixels in the *world coordinate system* into the *captured image coordinate system* using a perspective transformation [71]:

$$\lambda_i[u\ v\ 1]^\top = \mathbf{K}[\mathbf{R}_i|\mathbf{t}_i][X\ Y\ Z\ 1]^\top, \tag{18}$$

where $\mathbf{K}$, $\mathbf{R}_i$ and $\mathbf{t}_i$ are the intrinsic camera matrix, rotation matrix and translation vector respectively, (u, v) are the coordinates of a pixel in image I_i, (X, Y, Z) are the pixel coordinates of the point that is the backprojection of $I_i(u, v)$ into the *world coordinate system*, λ_i is a scaling factor of I_i.

We estimate $\mathbf{K}, \mathbf{R}_i$ and $\mathbf{t}_i$ with a closed-form solution [70], and minimize

$$\sum_{i=1}^{n}\sum_{j=1}^{L}\left\|\mathbf{m}_{i,j} - \hat{\mathbf{m}}(\mathbf{K}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_j)\right\|^2 \tag{19}$$

to refine them, where $\mathbf{m}_{i,j}$ is the intensity of a detected feature point in image I_i and $\hat{\mathbf{m}}$ is the projection of the world point $\mathbf{M}_j = [X_j\ Y_j\ Z_j]^\top$ in image I_i.

Given a checkerboard of size $(g \cdot w)$mm$\cdot(g \cdot h)$mm, and l mm away from the virtual camera in both I_{pp} and D_{pp}, the rotation matrix and translation vector of the virtual camera when 'capturing' the virtual *depth* pattern plane image D_{ppv}, denoted as $\mathbf{R}_{ppv}$ and $\mathbf{t}_{ppv}$ (as shown in Fig. 19) respectively, are given by:

$$\mathbf{R}_{ppv} = \mathbf{I}_3,\ \ \mathbf{t}_{ppv} = [\frac{g \cdot w}{2}\ \ \frac{g \cdot h}{2}\ \ l]^\top. \tag{20}$$

The virtual image coordinates function based on perspective transformation is given by:

$$\lambda_{ppv}[u_2\ v_2\ 1]^\top = \mathbf{K}\mathbf{R}_{pp}^{-1}\mathbf{K}^{-1}\lambda_{pp}[u_1\ v_1\ 1]^\top - \mathbf{K}\mathbf{R}_{pp}^{-1}\mathbf{t}_{pp} + \mathbf{K}\mathbf{t}_{ppv}, \tag{21}$$

where $\mathbf{R}_{pp}$ and $\mathbf{t}_{pp}$ are the rotation matrix and translation vector of the actual camera when capturing I_{pp}, $\lambda_{pp} = S_1/c_1$, c_1 is from:

$$[a_1\ b_1\ c_1]^\top = \mathbf{K}^{-1}[u_1\ v_1\ 1]^\top, \tag{22}$$

and the relationship between the actual depth value (in mm) S_1 and the observed disparity $D_{pp}(u_1, v_1)$ in D_{pp} is given by (see [72]):$S_1 = [-2.85 \times 10^{-6} D_{pp}(u_1, v_1) + 0.003]^{-1}$.

Similarly we have $S_2 = [-2.85\times10^{-6}D_{ppv}(u_2, v_2)+0.003]^{-1}$. Finally, the observed disparity of the point in the virtual image is given by: $D_{ppv}(u_2, v_2) = d = (0.003S_2 - 1)/(2.85 \times 10^{-6}S_2)$.

Appendix B
Bisection method

Following [44], we use the implicit form of the ellipse

$$E(x_n, y_n) = (\frac{x_n}{a})^2 + (\frac{y_n}{b})^2 - 1 = 0, \tag{23}$$

and calculate half of the gradient of $E(x_n, y_n)$, *i.e.*, the normal vector to (x_n, y_n), *i.e.*,

$$(u_n^{'}, v_n^{'}) - (x_n, y_n) = q\nabla\frac{E(x_n, y_n)}{2} = q(\frac{x_n}{a^2}, \frac{y_n}{b^2}), \tag{24}$$

or

$$u_n^{'} = x_n(1 + \frac{q}{a^2}), v_n^{'} = y_n(1 + \frac{q}{b^2}), \tag{25}$$

where q is a scalar. Without loss of generality $a \geq b$. With exception of the following four special cases for $s_\theta(\mathbf{o}_n)$:

$$s_\theta(\mathbf{o}_n) = \begin{cases} |\sqrt{a^2{u_n^{'}}^2 + b^2{v_n^{'}}^2} - a|, \text{if } a = b \\ a, \text{if } |u_n^{'} - a| < \varsigma, |v_n^{'} - b| < \varsigma \\ |u_n^{'} - a|, \text{if } |u_n^{'} - a| \geq \varsigma, |v_n^{'} - b| < \varsigma \\ |v_n^{'} - b|, \text{if } |u_n^{'} - a| < \varsigma, |v_n^{'} - b| \geq \varsigma \end{cases} \tag{26}$$

where $\varsigma > 0$ is a small tolerance, (25) can be solved for x_n and y_n as:

$$x_n = \frac{a^2u_n^{'}}{q + a^2}, y_n = \frac{b^2v_n^{'}}{q + b^2}. \tag{27}$$

Thus, we have

$$E(q) = (\frac{au_n^{'}}{q + a^2})^2 + (\frac{bv_n^{'}}{q + b^2})^2 - 1 = 0, \tag{28}$$

where $q \in [q_{\min}, q_{\max}], q_{\min} = -b^2 + bv_n^{'}, q_{\max} = -b^2 + \sqrt{a^2{u_n^{'}}^2 + b^2{v_n^{'}}^2}, E(q_{\min}) > 0, E(q_{\max}) < 0$ [44]. BS first examines the sign of $E(\frac{q_{\min}+q_{\max}}{2})$, then replaces $q_{\min}$ ($q_{\max}$) with $\frac{q_{\min}+q_{\max}}{2}$ if $E(q_{\min})$ ($E(q_{\max})$) has the same sign as $E(\frac{q_{\min}+q_{\max}}{2})$. Let all the subsequent intervals of q's be $[q^*_{\min}, q^*_{\max}]$. BS stops at $|q^*_{\max} - q^*_{\min}| < \tau$, where $\tau > 0$ is a small tolerance. We use the above BS procedure to determine $s_\theta(\mathbf{o}_n)$.

Appendix C
Nelder-Mead simplex method

NM starts from $V = \{V_1, \ldots, V_K\}$, the $(K + 1)$ points in K-dimensional space defining the initial simplex, for minimization of a function with k variables. Let $V_k = f_k(\theta)$. NM continuously updates V with three operations, naming, reflection, contraction, and expansion [47], until $\forall V_k, \sqrt{(V_k - \frac{1}{K}\sum_{k=1}^{K} V_k)^2/K} < \chi$, where χ is a small tolerance, *i.e.*, the minimum has been reached.

References

[1] A. Malhotra and D. P. White, "Obstructive sleep apnoea," *The Lancet*, vol. 360, no. 9328, pp. 237–245, Jul. 2002.

[2] P. Peppard et al., "Prospective study of the association between sleep-disordered breathing and hypertension," *The New England Journal of Medicine*, vol. 342, no. 19, pp. 1378–1384, May 2000.

[3] C. I. et al., *The AASM Manual for the Scoring of Sleep and Associated Events*. American Academy of Sleep Medicine, 2007.

[4] T. Wiegand et al., "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[5] J. Pang, G. Cheung, W. Hu, and O. C. Au, "Redefining self-similarity in natural images for denoising using graph signal gradient," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Siem Reap, Cambodia, Dec. 2014.

[6] J. Pang, G. Cheung, A. Ortega, and O. C. Au, "Optimal graph Laplacian regularization for natural image denoising," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015.

[7] D. I. Shuman et al., "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," in *IEEE Signal Processing Magazine*, vol. 30, no. 3, May 2013, pp. 83–98.

[8] R. R. Coifman, Y. Meyer, and V. Wickerhauser, "Wavelet analysis and signal processing," in *Wavelets and Their Applications*, M. B. Ruskai, Ed. Boston: Jones and Barlett, 1992, pp. 153–178.

[9] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 2008.

[10] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[11] D. D. Lee et al., "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, Oct. 1999.

[12] ——, "Algorithms for non-negative matrix factorization," in *Annual Conference on Neural Information Processing Systems*, T. K. Leen et al., Ed. MIT Press, 2001, pp. 556–562.

[13] M. W. Berry et al., "Algorithms and applications for approximate nonnegative matrix factorization," *Computational Statistics and Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[14] J. Behar et al., "A review of current sleep screening applications for smartphones," *Physiological Measurement*, vol. 34, no. 7, pp. R29–R46, Jun. 2013.

[15] D. S. Avalur, "Human breath detection using a microphone," Master's thesis, Faculty of Mathematics and Natural Sciences, University of Groningen, Aug. 2013.

[16] Z. Chen et al., "Unobtrusive sleep monitoring using smartphones," in *International Conference on Pervasive Computing Technologies for Healthcare and Workshops*, Venice, Italy, May 2013.

[17] N. Oliver and F. Flores-Mangas, "Healthgear: Automatic sleep apnea detection and monitoring with a mobile phone," *Journal of Communications*, vol. 2, no. 2, Mar. 2007.

[18] J. Behar et al., "SleepAp: An automated obstructive sleep apnoea screening application for smartphones," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 325–331, Jan. 2015.

[19] L. Jiang et al., "Automatic sleep monitoring system for home healthcare," in *IEEE-EMBS International Conference on Biomedical and Health Informatics*, Jan. 2012.

[20] D. C. Mack et al., "Development and preliminary validation of heart rate and breathing rate detection using a passive, ballistocardiography-based sleep monitoring system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 1, pp. 111–120, Jan. 2009.

[21] K. Malakuti and A. Albu, "Towards an intelligent bed sensor: Non-intrusive monitoring of sleep irregularities with computer vision techniques," in *International Conference on Pattern Recognition*, Istanbul, Turkey, Aug. 2010.

[22] J. Paalasmaa et al., "Unobtrusive online monitoring of sleep at home," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2012.

[23] N. Patwari et al., "Monitoring breathing via signal strength in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1774–1786, Aug. 2014.

[24] M. Martinez et al., "Breath rate monitoring during sleep using near-IR imagery and PCA," in *International Conference on Pattern Recognition*, Tsukuba, Japan, Nov. 2012.

[25] A. Loblaw et al., "Remote respiratory sensing with an infrared camera using the Kinect$^{\text{TM}}$ infrared projector," in *World Congress in Computer Science, Computer Engineering, & Applied Computing*, 2013.

[26] B. Krüger et al., "Sleep detection using de-identified depth data," *Journal of Mobile Multimedia*, vol. 10, no. 3&4, pp. 327–342, Dec. 2014.

[27] D. Falie et al., "Respiratory motion visualization and the sleep apnea diagnosis with the time of flight (ToF) camera," in *WSEAS International Conference on Visualization, Imaging and Simulation*, Bucharest, Romania, Nov. 2008.

[28] M.-C. Yu et al., "Multiparameter sleep monitoring using a depth camera," in *Biomedical Engineering Systems and Technologies*, J. Gabriel et al., Ed. Springer, 2013, vol. 357, pp. 311–325.

[29] C.-W. Wang et al, "Unconstrained video monitoring of breathing behavior and application to diagnosis of sleep apnea," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 2, pp. 396–404, Feb. 2014.

[30] M. W. Lee and R. Nevatia, "Body part detection for human pose estimation and tracking," in *IEEE Workshop on Motion and Video Computing*, Austin, TX, Feb. 2007.

[31] J. Shotton et al., "Real-time human pose recognition in parts from single depth images," in *IEEE Conference on Computer Vision and Pattern Recognition*, Collorado Springs, CO, Jun. 2011.

[32] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake, "Efficient human pose estimation from single depth images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821–2840, Dec. 2013.

[33] J. Yu, Y. Rui, Y. Y. Tang, and D. Tao, "High-order distance-based multiview stochastic learning in image classification," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2431–2442, Dec. 2014.

[34] J. Yu, D. Tao, J. Li, and J. Cheng, "Semantic preserving distance metric learning and applications," *Information Sciences*, vol. 281, pp. 674 – 686, Oct. 2014.

[35] M. Madadi et al., "Multi-part body segmentation based on depth maps for soft biometry analysis," *Pattern Recognition Letters*, vol. 56, pp. 14–21, Apr. 2015.

[36] V. Metsis et al., "Non-invasive analysis of sleep patterns via multimodal sensor input," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 19–26, Jan. 2014.

[37] L.-C.-L. Chen et al., "A sleep monitoring system based on audio, video and depth information for detecting sleep events," in *IEEE International Conference on Multimedia & Expo*, Chengdu, China, Jul. 2014.

[38] J. Lee et al, "Sleep monitoring system using kinect sensor," *International Journal of Distributed Sensor Networks*, vol. 2015, Apr. 2015.

[39] F. Centonze et al., "Feature extraction using ms kinect and data fusion in analysis of sleep disorders," in *International Workshop on Computational Intelligence for Multimedia Understanding*, Prague, Czech Republic, Oct. 2015.

[40] C. Yang, G. Cheung, K. Chan, and V. Stankovic, "Sleep monitoring via depth video recording & analysis," in *IEEE International Workshop on Hot Topics in 3D*, Chengdu, China, Jul. 2014.

[41] C. Yang, Y. Mao, G. Cheung, V. Stankovic, and K. Chan, "Graph-based depth video denoising and event detection for sleep monitoring," in *IEEE International Workshop on Multimedia Signal Processing*, Jakarta, Indonesia, Sept. 2014.

[42] R. N. Khushaba et al., "Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 121–131, Jan. 2011.

[43] R. L. Burden and J. D. Faires, *Numerical Analysis: 4th Edition*. Boston, MA, USA: PWS Publishing Co., 1989.

[44] D. H. Eberly, "Distance from a point to an ellipse, an ellipsoid, or a hyperellipsoid," Geometric Tools, LLC, Tech. Rep., 1998.

[45] W. H. Press et al., *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007.

[46] D. H. Eberly, *3D Game Engine Design, Second Edition: A Practical Approach to Real-Time Computer Graphics*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2006.

[47] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, pp. 308–313, 1965.

[48] J. C. Lagarias et al., "Convergence properties of the nelder–mead simplex method in low dimensions," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, Dec. 1998.

[49] K. C. Pohlmann, *Principles of Digital Audio*, 6th ed. McGraw-Hill Professional, 2010.

[50] Y.-W. Huang et al., "Survey on block matching motion estimation algorithms and architectures with new results," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 42, pp. 297–320, Mar. 2006.

[51] W. Hu, X. Li, G. Cheung, and O. Au, "Depth map denoising using graph-based transform and group sparsity," in *IEEE International Workshop on Multimedia Signal Processing*, Pula, Italy, Oct. 2013.

[52] W. Hu, G. Cheung, X. Li, and O. Au, "Depth map compression using multi-resolution graph-based transform for depth-image-based rendering," in *IEEE International Conference on Image Processing*, Orlando, FL, Sept. 2012.

[53] W. Hu, G. Cheung, A. Ortega, and O. Au, "Multi-resolution graph Fourier transform for compression of piecewise smooth images," in *IEEE Transactions on Image Processing*, vol. 24, no. 1, Jan. 2015, pp. 419–433.

[54] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE International Conference on Computer Vision*, Bombay, India, 1998.

[55] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[56] R. Safaee-Rad et al., "Accurate parameter estimation of quadratic curves from grey-level images," *CVGIP: Image Understanding*, vol. 54, no. 2, pp. 259–274, Sept. 1991.

[57] W. Gander et al., "Least-square fitting of circles and ellipses," *BIT Numerical Mathematics*, vol. 34, no. 4, pp. 558–578, Dec. 1994.

[58] P. Rosin, "Analysing error of fit functions for ellipses," *Pattern Recognition Letters*, vol. 17, no. 14, pp. 1461–1470, 1996.

[59] R. B. Berry et al., "Rules for scoring respiratory events in sleep: Update of the 2007 AASM manual for the scoring of sleep and associated events: Deliberations of the sleep apnea definitions task force of the american academy of sleep medicine," *Journal of Clinical Sleep Medicine*, vol. 8, no. 5, pp. 597–619, Oct. 2012.

[60] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[61] R. R. Coifman et al., "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 713–718, Mar. 1992.

[62] C.-W. Hsu et al., "A practical guide to support vector classification," Department of Computer Science, National Taiwan University, Tech. Rep., 2003.

[63] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, Mar. 2003.

[64] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, no. 1-2, pp. 273–324, Dec. 1997.

[65] S. Matyunin et al., "Temporal filtering for depth maps generated by kinect depth camera," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Antalya, Turkey, 2011.

[66] D. Min et al., "Depth video enhancement based on weighted mode filtering," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1176–1190, Mar. 2012.

[67] S. H. Chan et al., "An augmented lagrangian method for total variation video restoration," *IEEE Transactions on Image Processing*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

[68] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[69] T. Giannakopoulos and A. Pikrakis, *Introduction to Audio Analysis, A MATLAB® Approach*. Academic Press, 2014.

[70] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[71] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[72] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, Feb. 2012.