November 2020

PASSIVE SLEEP DETECTION

Kathy Chen

Tajinder Gadh

Marc Stogaitis

PASSIVE SLEEP DETECTION

ABSTRACT

A system is described that enables a computing system (e.g., a mobile phone, a smartwatch, a tablet computer, etc.) to passively detect a user's sleep duration. That is, without a user configuring the computing system into a sleep mode or otherwise inputting a sleep duration, the computing system may, after receiving explicit permission from the user, monitor various contextual signals to automatically determine the user's sleep duration. The computing system may passively capture various data using sensors (e.g., accelerometers, ambient light sensors, microphones, etc.) in the computing system and analyze the captured data to estimate a user's sleep duration. For example, the computing system may analyze accelerometer data to determine when a user is moving and how much the user is moving, analyze audio data captured by a microphone to determine if the audio captured is indicative of sleep, and/or analyze ambient light data to determine ambient light conditions. Such sensor data may be periodically generated and analyzed to generate sleep information for s series of time intervals. Based on the analysis of such sensor data, the computing system may determine whether the user was asleep when the computing system generated the sensor data. In some examples, the computing device may further classify sleep stages (e.g., rapid eye movement (REM) stage, light sleep stage, deep sleep stage, etc.) using the generated sensor data (e.g., classify sleep stages using user's breathing rate, heart rates, or movement, etc.).

DESCRIPTION

Sleep is a critical aspect of people's well-being and sleep duration is an important indicator of a person's health. Existing sleep assessment methods are often expensive, timeconsuming, and often require professional assistance. As such, it would be desirable for a computing system to be able to determine a user's sleep duration. Techniques of this disclosure may enable a computing system to determine a user's sleep using generated sensor data. Rather than applying a general default sleep detection, the computing system may request authorization from a user to generate sensor data and, after receiving explicit authorization from the user, generate and analyze the generated sensor data, and determine a user's sleep duration based on the generated sensor data.

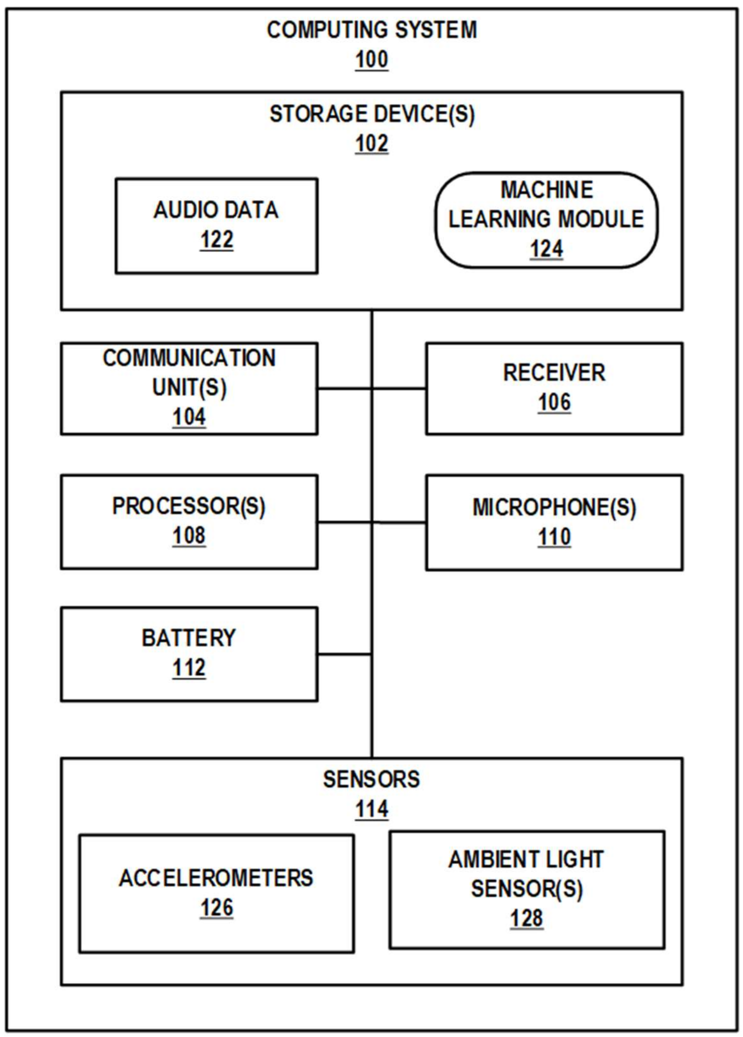

FIG. 1

FIG. 1 illustrates an example computing system 100 that is configured to determine a user's sleep duration. In the example of FIG. 1, computing system 100 includes one or more storage devices 102, one or more communication units 104, a receiver 106, one or more processors 108, one or more microphones 110, a battery 112, and a set of sensors 114, and one or more communication channels 216. Computing system 100 may include, or otherwise be included in, a mobile device (e.g., smartphone), a smartwatch, a tablet computer, a laptop computer, or any other type of suitable audio capture device.

In the example of FIG. 1, sensors 114 include one or more accelerometers 126 for determining the motion of computing system 100 and one or more ambient light sensors 128 for determining the ambient light level around computing system 100. Sensors 114 may further include one or more additional sensors, such as a gyroscope, a magnetometer, environmental temperature sensors, and/or other types of sensors. Storage device(s) 102 store audio data 122 and machine learning module 124. For instance, storage device(s) 102 may store audio data 122 obtained from microphone(s) 110 and machine learning module 124 may determine a user's sleep duration based on audio data 102.

In various instances, a user of computing system 100 may wish to track the user's sleep patterns and duration. In such instances, the user may provide explicit permission to allow computing system 100 to capture and analyze audio data and sensor data generated by computing system 100. After receiving explicit user permission, computing system 100 may capture data from microphone(s) 110 and sensors 114 for a configurable duration. In some examples, computing system 100 may obtain audio data from microphone(s) 110 every six minutes, nine minutes, twenty minutes, etc. or may obtain audio data from microphone(s) 110 in an aperiodic fashion. For example, machine learning module 124 may control microphone(s) 110 to perform random audio recording at random times during a set time period (e.g., randomly throughout each day).

Computing system 100 may then segment the captured data to data indicating a sleep event and data indicating an active event. For example, a machine learning module 124 may be used to segment the captured audio data 122 into silent segments and non-silent segments, wherein silent segments are audio segments generated during sleep events and non-silent segments are audio segments generated during active events. Various algorithms may be used for sleep event segmentation. In one example, machine learning module 124 uses a Hidden Semi-Markov model to segment audio data 122 into silent segments and non-silent segments.

In some examples, machine learning module 124 may segment audio data 122 based on movement occurrence in the audio data. For example, machine learning module 124 may segment audio data 122 by calculating a metric score for each time unit (e.g., 5 seconds, 10 seconds, 30 seconds, etc.), wherein the metric score may be calculated based on the detected movement of computing system 100. Audio data with sounds indicating movement for a relatively long period of time may be assigned with a relatively higher metric score than audio data with sounds indicating movement for a relatively shorter period of time. The machine learning module 124 may then compare the calculated metric scores with a pre-defined metric threshold value. For instance, in response to determining that the calculated metric scores for certain time units have not satisfied the pre-define metric threshold value, machine learning module 124 may segment audio data 122 into silent segments and non-silent segments based on this determination.

In some examples, machine learning module 124 may segment audio data 122 based on sound occurrence in the audio data. In particular, machine learning module 124 may segment audio data 122 based on a change of average frequency between two time units (e.g., 5 seconds, 10 seconds, 30 seconds, etc.). For example, breathing sound frequency may be relatively lower than voice frequency. Thus, a sudden change of average frequency between two time units may be used to segment audio data 122 into silent segments and non-silent segments.

In some examples, machine learning module 124 may segment audio data 122 based on ambient light level determined based on ambient light data captured by ambient light sensors 128. For example, machine learning module 124 may segment audio data 122 by calculating a metric score for each time unit (e.g., 5 seconds, 10 seconds, 30 seconds, etc.), wherein the metric score may be calculated based on detected ambient light levels for each segment. Audio data with a relatively high ambient light level may be assigned with a relatively higher metric score than audio data with a relatively low ambient light level. The machine learning module 124 may then compare the calculated metric scores with a pre-defined metric threshold ambient light level.

In some examples, machine learning module 124 may segment audio data 122 based on accelerometer data captured by accelerometers 126. For example, computing system 100 may be in a completely static state during a user's sleeping hours, and the user may pick up computing system 100 after he wake (e.g., to turn off an alarm clock of computing system 100). The machine learning module 124 may then use the accelerometer data captured by accelerometers 126 to segment audio data 122 into silent segments and non-silent segments.

Machine learning module 124 may further extract sound features from non-silent segments and filter out false-positive non-silent segments (e.g., non-silent segments that include snoring, yawning, background noise, etc.) based on the extracted sound features. In some examples, machine learning module 124 may extract the frequency of a sound, the amplitude of the sound, and other sound features from a non-silent segment to determine whether the non-silent segment should be reclassified as a silent segment. For example, machine learning module 124 may determine the extracted frequency of sound and amplitude of sound from the non-silent segment matches the frequency and amplitude of a stored snoring sound sample, and may reclassify the non-silent segment as a silent segment as snoring sound may indicate a sleep event.

In some examples, machine learning module 124 may filter out false-positive non-silent segments (e.g., non-silent segments that include train noise) based on both of the extracted sound features and/or motion sensor data. For example, machine learning module 124 may determine the extracted frequency of sound and amplitude of sound from the non-silent segment matches the frequency and amplitude of a stored train noise sound sample and/or may determine that a accelerometer data detected by accelerometers 126 based on the accelerometer data matches a stored train vibration data. Based on the determination, machine learning module 124 may reclassify the non-silent segment as a silent segment as the train noise may not indicate an active event.

Machine learning module 124 may then estimate a user's sleep duration based on the silent segments. For example, machine learning module 124 may estimate the user's sleep duration by aggregating time data of the silent segments.

In some examples, machine learning module 124 may further classify a user's sleep stages using various captured data. For example, machine learning module 124 may classify a user's sleep stages (e.g., rapid eye movement (REM) stage, light sleep stage, deep sleep stage, etc.) using the user's breathing rate, heart rates, and movement. For example, a relatively slow breathing rate may indicate a user entered deeper sleep while a relatively fast breathing rate may indicate the user entered lighter sleep.

Machine learning module 124 may estimate a user's breathing rate using envelope detection. For example, a user may sleep with his phone next to him (e.g., 1 foot away from the user, such as on a nightstand), and microphones included in the phone may capture the respiratory sound of the user and estimate the user's breathing rate based on captured audio data. Machine learning module 124 may determine inhales and exhales from captured respiratory sound and may cause a change in signal amplitude based on detecting inhales or exhales. Respiration curve, which represents the user's breathing rate, may then be obtained through envelope detection of the signal.

It is noted that the techniques of this disclosure may be combined with any other suitable technique or combination of techniques. As one example, the techniques of this disclosure may be combined with the techniques described in US Patent Application Publication 2016/0272718 A1. As another example, the techniques of this disclosure may be combined with the techniques described by Jun-Ki Min, Afsaneh Doryab, Jason Wiese, Shahriyar Amini, John Zimmerman, and Jason I. Hong, "Toss 'N' turn: Smartphone as sleep and sleep quality detector," available at https://www.cs.utah.edu/~wiese/publications/tossnturn.pdf. As another example, the techniques of this disclosure may be combined with the techniques described by Ruth Ravichandran, Elliot Saba, Ke-Yu Chen, Mayank Goel, Sidhant Gupta, and Shwetak N. Patel, "WiBreathe: Estimating respiration rate using wireless signals in natural settings in the home," available at https://ieeexplore.ieee.org/document/7146519.