METHOD AND DEVICE FOR TONGUE IMAGE EXTRACTION AND COMPUTER READABLE STORAGE MEDIUM

CROSS-REFERENCE TO RELATED APPLICATION(S)

[0001] This application claims priority to Chinese Patent Application No. 201910733855.8 filed August 09, 2019, entitled "METHOD AND DEVICE FOR TONGUE IMAGE EXTRACTION AND COMPUTER READABLE STORAGE MEDIUM", the entire disclosure of which is hereby incorporated by reference.

TECHNICAL FIELD

[0002] This disclosure relates to a method and a device for tongue image extraction and a computer readable storage medium.

BACKGROUND

[0003] In related art, methods for tongue image detection are usually based on object detection. For example, a sliding window is used to slide horizontally and vertically on an image respectively, and spatial features are extracted from an object in the sliding window through a convolutional neural network (CNN) model, and the extracted spatial features are classified with a support vector machine (SVM) classifier, to determine whether in the sliding window is a tongue image. The coordinates of four corner points of the sliding window are output, and a position of the tongue image is calibrated with the coordinates of the four corner points. However, due to large differences in size, angle, and posture of tongues in different images, size of a target box is uncertain. This requires multiple times of sliding for recognition through target boxes of various sizes, leading to a certain complexity of object detection.

[0004] Therefore, how to quickly obtain a correct, complete, and clear tongue image is a problem to be solved.

SUMMARY

[0005] Implementations of this application provide a method for tongue image extraction. The method is applicable to an electronic device and includes the following.

[0006] At S110, training images containing tongue are converted into a matrix V, where

all non-negative gray-values of one image correspond to one column of the matrix V, and train with a local non-negative matrix factorization (LNMF) algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH. The non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, where the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and the weight matrix H has a dimension of r*m and each column is encoded.

[0007] At S120, an embedded hidden Markov model (EHMM) is used to recognize whether a test image(s) contains face, the training images and the test image are projected to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, a nearest neighbor criterion is used to obtain similarities among the feature coefficients corresponding to the training images and the test image, and features of the test image representing tongue and having similarities higher than a similarity threshold are extracted to be tongue features.

[0008] At S130, after projection, a feature region with the tongue features and a non-feature region without the tongue features are identified with different labels respectively, where a set of labels corresponds to boundary information of the feature region, extreme values in up, down, left, and right directions of the boundary information are extracted, and a bounding box defining the feature region is determined.

[0009] This application further provides a device for tongue image extraction. The device for tongue image extraction includes a matrix factorization module, a tongue features extracting module, and a tongue image segmenting module.

[0010] The matrix factorization module is configured to convert training images containing tongue into a matrix V, where all non-negative gray-values of one image correspond to one column of the matrix V, and train with an LNMF algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH. The non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, where the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and the weight matrix H has a dimension of r*m and each column is encoded.

[0011] The tongue features extracting module is configured to use an EHMM to recognize whether a test image contains face, project the training images and the test image to the non-negative subspace respectively to obtain feature coefficients when the test image

contains face, use a nearest neighbor criterion to obtain similarities among the feature coefficients corresponding to the training images and the test image, and extract features of the test image representing tongue and having similarities higher than a similarity threshold to be tongue features.

[0012] The tongue image segmenting module is configured to identify, with different labels, a feature region with the tongue features and a non-feature region without the tongue features respectively, where a set of labels corresponds to boundary information of the feature region, extract extreme values in up, down, left, and right directions of the boundary information, and determine a bounding box defining the feature region.

[0013] This application further provides an electronic device. The electronic device includes a processor and a memory configured to store a program for tongue image extraction. When executed by the processor, the program is configured to implement the following.

[0014] At S110, training images containing tongue are converted into a matrix V, where all non-negative gray-values of one image correspond to one column of the matrix V, and train with an LNMF algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH. The non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, where the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and the weight matrix H has a dimension of r*m and each column is encoded.

[0015] At S120, an EHMM is used to recognize whether a test image contains face, the training images and the test image are projected to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, a nearest neighbor criterion is used to obtain similarities among the feature coefficients corresponding to the training images and the test image, and features of the test image representing tongue and having similarities higher than a similarity threshold are extracted to be tongue features.

[0016] At S130, after projection, a feature region with the tongue features and a non-feature region without the tongue features are identified with different labels respectively, where a set of labels corresponds to boundary information of the feature region, extreme values in up, down, left, and right directions of the boundary information are extracted, and a bounding box defining the feature region is determined.

[0017] This application further provides a non-transitory computer readable storage medium. The non-transitory computer readable storage medium stores a computer program.

The computer program includes program instructions. When executed by a processor, the program instructions are configured to implement the method for tongue image extraction of the above.

[0018] The details of one or more implementations of this application are set forth in the following drawings and description. Other features and advantages of this application will be apparent from the description, drawings and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0019] Implementations of this application are described in conjunction with the following drawings, such that the above features and technical advantages of this application will become clearer and easier to be understood.

[0020] FIG. 1 is a schematic flowchart illustrating a method for tongue image extraction according to implementations.

[0021] FIG. 2 is a first schematic flowchart illustrating super states and embedded states of an embedded hidden Markov model (EHMM) corresponding to image slices according to implementations.

[0022] FIG. 3 is a second schematic flowchart illustrating super states and embedded states of an EHMM corresponding to image slices according to implementations.

[0023] FIG. 4 is a third schematic flowchart illustrating super states and embedded states of an EHMM corresponding to image slices according to implementations.

[0024] FIG. 5 is a schematic diagram illustrating a hardware architecture of an electronic device according to implementations.

[0025] FIG. 6 is a schematic diagram illustrating program modules of a program for tongue image extraction according to implementations.

[0026] FIG. 7 is a schematic diagram illustrating adjustment of a bounding box of a linear regression model according to implementations.

DETAILED DESCRIPTION

[0027] A method and a device for tongue image extraction and a non-transitory computer readable storage medium of implementations of this application are described below with reference to the drawings.

[0028] In at least one implementation, FIG. 1 is a schematic flowchart illustrating a method for tongue image extraction according to implementations. The method is applicable to an electronic device and begins at S110.

[0029] At S110, train with a local non-negative matrix factorization (LNMF) algorithm, to obtain a feature base image(s) of a different dimension. For example, 1000 tongue images (that is, images each containing tongue and reflecting shape and color of the tongue) are used as a set of training images, and the tongue images have been annotated in advance. In an example, the tongue images can be compressed first, for example, compressed to 56*64 pixels, and the tongue images are de-averaged and normalized. The LNMF algorithm is used for training, to obtain the feature base image of a different dimension. The feature base image refers to a non-negative feature matrix W representing tongue features (i.e., features of tongue) and the non-negative feature matrix W constitutes a non-negative subspace.

[0030] The LNMF is improvement of NMF. The LNMF algorithm refers to factorizing a matrix V corresponding to the training images into a product of the non-negative feature matrix W and a weight matrix H, that is, V=WH.

[0031] The matrix V has a dimension of n*m, where V= (V1, V2, ……, Vm). All non-negative gray-values of one image correspond to one column of the matrix V, and data in the matrix V is gray-values of the training images.

[0032] The non-negative feature matrix W has a dimension of n*r, and r columns are a base image.

[0033] The weight matrix H has a dimension of r*m, where each column of the weight matrix H is encoded and corresponds to one tongue image in the matrix V. Therefore, one training image can be expressed as a linear combination of the base image.

[0034] At S120, an embedded hidden Markov model (EHMM) is used to recognize whether a test image(s) contains face, and feature extraction is performed on the test image when the test image contains face. The non-negative feature matrix W representing tongue features constitutes the non-negative subspace. The training images and the test image are respectively projected to the non-negative subspace obtained by the set of training images, to obtain feature coefficients. A nearest neighbor criterion is used to obtain similarities among the feature coefficients corresponding to the training images and the test image. Features of the test image representing tongue and having similarities higher than a set threshold are extracted to be tongue features. In this way, an image with the tongue features can be selected from the test image. The tongue features include shape, angle, color, state of coating of the tongue, and positional relationship between the tongue and facial organs.

[0035] At S130, the test image is projected to the non-negative subspace, where the projection is equivalent to transforming the test image into the non-negative subspace, and

the test image is still an image which is composed of learned features. After projection, a feature region with the tongue features and a non-feature region without the tongue features are identified with different labels respectively, such that the feature region with the tongue features is segmented from the test image. A set of labels corresponds to boundary information of the feature region, extreme values in up, down, left, and right directions of the boundary information are extracted, and a minimum bounding box defining the feature region is determined. The reason for using the minimum bounding box herein is that linear regression will be used to adjust a position of the minimum bounding box to eliminate or reduce a position error. The non-feature region and the feature region have different labels, for example, the non-feature region is denoted by "0" and the feature region is denoted by "non-zero". According to "0" and "non-zero", an image region representing the tongue features can be segmented from the test image by using the bounding box. Furthermore, the method includes S140. At S140, support vector machine (SVM) classifiers are used to classify the features extracted from the test image, where the extracted features are sent to k SVM classifiers for recognition and k is equal to the number of classes. In an example, the classes include "tongue" and "non-tongue". In another example, the classes include pathological conditions of the tongue. The physical conditions of a person may include damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis. For the k SVM classifiers, a class with the highest score is used as a result for classification.

[0036] In at least one implementation, the method further includes S150. At S150, a linear regression model is used to adjust the position of the bounding box on the tongue image. For each class, for example, damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis, train one linear regression model, where input is features of the image in the bounding box and output is translation (left-right translation and up-down translation) values and a zoom value of the bounding box. The linear regression model is used to calculate the translation values and the zoom value of the bounding box and a loss function is used to constrain the position error of the bounding box, to adjust the bounding box to move to a suitable position.

[0037] As illustrated in FIG. 7, a linear regression model involves an original value (or called proposal) of a given position $P=(P_x, P_y, P_w, P_h)$, where P_x, P_y denotes coordinates of a bounding box, and P_w and P_h respectively denote width and height of the bounding box. A mapping f is obtained through machine learning, such that $f(P_x, P_y, P_w, P_h) = (\widehat{G_x}, \widehat{G_y}, \widehat{G_w}, \widehat{G_h})$, and

a predicted value of the position: $(\widehat{G_x}, \widehat{G_y}, \widehat{G_w}, \widehat{G_h}) \approx$ a true value (or called ground truth) of the position: (G_x, G_y, G_w, G_h).

[0038] Assume that translation is $(\Delta x, \Delta y)$, $\Delta x = P_w d_x(P)$, $\Delta y = P_h d_y(P)$, then

$$\widehat{G_x} = P_w d_x(P) + P_x \quad (1)$$

$$\widehat{G_y} = P_h d_y(P) + P_y \quad (2).$$

[0039] Assume that zoom of scale is (S_w, S_h), $S_w = \exp(d_w(P))$, $S_h = \exp(d_h(P))$, then

$$\widehat{G_w} = P_w \exp(d_w(P)) \quad (3)$$

$$\widehat{G_h} = P_h \exp(d_h(P)) \quad (4).$$

[0040] The bounding box regression refers to obtaining four accurate transformation values $d_x(P)$, $d_y(P)$, $d_w(P)$, $d_h(P)$ through learning.

[0041] Input is $P=(P_x, P_y, P_w, P_h)$ and output is the predicted value of the position $\widehat{G}$. To transform the original value of the position to the true value G of the position, a true transformation value $t_* = (t_x, t_y, t_w, t_h)$ is required, where a true translation amount is (t_x, t_y) and width and height of true zoom of scale are (t_w, t_h), where

$$t_x = (G_x - P_x)/P_w \quad (5)$$
$$t_y = (G_y - P_y)/P_h \quad (6)$$
$$t_w = \log(G_w/P_w) \quad (7)$$
$$t_h = \log(G_h/P_h) \quad (8)$$

[0042] An objective function $d_*(P) = w_*^T K(P)$ is constructed, where w_* is a variable to-be-learned (* denotes x, y, w, h, and set one object function for each variable), and $d_*(P)$ is a predicted transformation value. K(P) is a feature vector (i.e., eigenvector) corresponding to a feature region. A difference between the predicted transformation value and the true transformation value $t_* = (t_x, t_y, t_w, t_h)$ should be minimum. A loss function is constructed and minimized:

$$\text{argmin}_{\text{Loss}} = \text{argmin}\left(\sum_{i=1}^{N}\left(t_*^i - w_*^T K(P^i)\right)^2\right) \quad (9),$$

where, i is the i-th iteration; and

N is the number of samples.

[0043] Through training of some samples, the loss function is minimized, to obtain w_*, that is, obtain $d_*(P)$, i.e., values of $d_x(P)$, $d_y(P)$, $d_w(P)$, $d_h(P)$.

[0044] In an example, NMF is a projection method based on local subspace. Features extracted by the NMF algorithm are global features, and no restriction is made on locality of a feature space. In order to enhance localization of principal components of the feature matrix W, LNMF emphasizes localization of basic feature components in original image decomposition. The formula of the LNMF algorithm is as follows:

construct an objective function:

$$D(V \| WH) = \sum_{i,j} \left[V_{ij} \log \frac{V_{ij}}{(WH)_{ij}} - V_{ij} - (WH)_{ij} \right] + \alpha \sum_{ij} (W^T W)_{ij} - \beta \sum_{ij} (HH^T)$$

where α and β are positive constants;

$$V, W, H \geq 0;$$

$\forall j: \|W_j\| = 1$, W_j represents the j-th column vector of the feature matrix W, meaning that each column of the feature matrix W is normilized;

$V = [V_1, V_2 \cdots V_i \cdots V_m]$ represents a set of m training images, clolumn vector V_i represents i-th training image, V_{ij} represents the j-th gray-value of the i-th training image, each training image has a size of n, and V has a size of n*m.

$W = [W_1, W_2, \cdots W_j \cdots W_r]$ is the feature matrix, having a size of n*r;

$H = [H_1, H_2 \cdots H_j \cdots H_m]$ is the weitht matrix, H_j is j-th column vector of H, having a size of r*m.

W and H are iteratively updated through the following formula, to achieve minimum of the objective function:

$$H_{jl} = \sqrt{H_{jl} \sum_i V_{il} \frac{W_{ij}}{\sum_j W_{ij} H_{jl}}}$$

$$W_{ij} = \frac{W_{il} \sum_l V_{il} \frac{H_{jl}}{\sum_i W_{il} H_{jl}}}{\sum_l H_{jl}}$$

$$W_{ij} = \frac{W_{ij}}{\sum_{ij} W_{ij}}$$

where i=1, 2, …, m; j=1, 2, …, r; l=1, 2, …, n, in each iteration, W and H are kept non-negative.

[0045] In at least one implementation, the training images and the test image are binarized. The binarization refers to setting gray-values of pixels of the training images and the test image to be 0 or 255, that is, a process of presenting the entire image with clear black and white effect. As such, a tongue region can be acquired more accurately. In an example,

the gray-values of the pixels of the training images and the test image are set to be 0 or 255 according to a set gray-value threshold. The gray-value threshold can be a median of 0~255, where gray-values less than the gray-value threshold are set to be 0 and gray-values greater than the gray-value threshold are set to be 255.

[0046] In at least one example, the test image is first classified using the EHMM. For example, the test image is classified into two classes: “face contained” and “no face contained”. Thus, accuracy of recognition can be improved. The specific classification process is as follows.

[0047] Multiple feature points of the face are selected to form a feature sequence.

[0048] The test image is input into the EHMM to scan the test image from top to bottom and from left to right through a sliding window (or moving window). The sliding window first scans from left to right and each window obtains a group of eigenvectors through scanning, which is a type of feature extraction on a face region. After the sliding window obtains the eigenvectors, it moves to the right at a fixed distance to continue the feature extraction. When moving to the right side of the image, the sliding window switches to the next line to continue scanning from left to right. When the sliding window moves to the bottom right of the image, the whole scanning process ends and multiple groups of eigenvectors are obtained, where the multiple groups of eigenvectors form a sequence of observations (or observation values).

[0049] The EHMM includes a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction. Each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction. For the EHMM, the image is scanned from left to right and from top to bottom through a window of a fixed-size. The facial features can correspond to the super states from top to bottom and the embedded states from left to right. As illustrated in FIG. 2, slices of images (or image slices) corresponding to the longitudinal super states are forehead region, eye region, nose region, mouth region, and chin region. From top to bottom, these regions have fixed relationship in terms of position, which is common to human faces. Therefore, personality of the face in the vertical direction is reflected by characteristics of each super state (that is, each region) and relationship among the super states. From left to right, the face is divided into left face, left eye, middle of the two eyes, right eye, and right face. These regions also have fixed relationship in terms of position. Therefore, personality of the face in the horizontal direction is reflected by each embedded state itself and mutual connection

among the embedded states.

[0050] A forward algorithm is used to derive a similar probability between the sequence of observations and the feature sequence containing the multiple feature points of the face. Recognize that the test image contains face when the similar probability is greater than a determination threshold.

[0051] In at least one example, the EHMM is trained as follows.

[0052] 1) EHMM modeling: the EHMM can be defined as a ternary formula of $\lambda = (P_0, A_0, \Lambda)$, where $\Lambda = \Lambda^1, \Lambda^2, \cdots, \Lambda^{N_0}$, and basic elements of the EHMM include:

(1) an initial probability of a super state: $P_0 = \pi_{0,i}$, where $\pi_{0,i}$ is a probability of super state *i* at time=0, $1 \leq i \leq N_0$, and N_0 represents the number of super states;

(2) a transition matrix $A_0 = a_0, ij$, where a_0, ij is a probability that the transition matrix changes from super state *i* to super state *j*. In the EHMM from left to right, the only allowed transition is from a current state to a next state. Therefore, a probability from an original state to a previous state is 0;

(3) $\Lambda^k = (P_1^k A_1^k B^k)$ represents a parameter set of the k-th super state, where $1 \leq k \leq N_0$;

P_1^k is an initial probability distribution of an embedded state;

A_1^k a transition probability matrix of an embedded state;

B^k represents an observation probability matrix, $b_j^k(O_{t_0,t_1})$ represents a probability of embedded state j of super state k generating observation O_{t_0,t_1}, and two variables of O_{t_0,t_1} respectively correspond vertical and horizontal dimensions:

$$b_j^k(O_{t_0,t_1}) = \sum_{m=1}^{M_j^k} C_{jm}^k N(O_{t_0,t_1}, \mu_{jm}^k, U_{jm}^k),$$

M_j^k represents the number of mixed Gaussians;

C_{jm}^k is a mixing coefficient of the m-th mixing term of embedded state j of super state k;

$N(O_{t_0,t_1}, \mu_{jm}^k, U_{jm}^k)$ is Gaussian density of mean vector μ_{jm}^k and covariance matrix $\bigcup_{jm}^k$.

[0053] 2) Image segmentation: the training image is uniformly divided, and a sequence of observations obtained from the image is evenly divided into N_0 longitudinal slices

corresponding to longitudinal super states. Each longitudinal slice can be sliced into multiple embedded states from left to right.

[0054] 3) Parameter initialization: after segmentation, initial values of parameters of the model are obtained through an initialization probability and a transition probability of each state. Each state of the EHMM uses K-means clustering to calculate an observation probability. K is the number of Gaussian distributions in each state. For observation vectors extracted in all embedded states, a Gaussian mixture model can be used to explain an observation probability density function. The state initialization rule of each super state is as follows: an initialization probability of each first state of EHMM is specified as 1.0 and an initialization probability of other states is 0.

[0055] 4) Embedded Viterbi segmentation: after a first iteration, a dual embedded Viterbi algorithm is used to replace uniform segmentation, and through new segmentation and event frequency counting, a new set of initialization and transition probabilities are determined.

[0056] 5) Segmentation of K-means clustering: according to the segmentation result of step 4), K-means clustering is used to calculate an observation vector corresponding to a new state, and a new observation probability density function. In the next iteration, these values will be used as initial values for a new round of dual embedded Viterbi segmentation.

[0057] 6) Repeat steps 4) and 5) until change in the continuous iteration is less than a set convergence threshold.

[0058] In at least one example, as illustrated in FIG. 3, slices of the face in the longitudinal direction also include a hair region. Although not everyone has hair, the hair region provides additional features of the face and does help to recognize the face more accurately. The calculation process is similar as the above process, which is not repeated herein.

[0059] In at least one example, in the above, the entire face is divided into slice images in the vertical direction to correspond to the super states and is divided into slice images in the horizontal direction to correspond to the embedded states. However, since this application focuses on tongue image extraction, for face recognition, part of the slices in the vertical direction can be used to correspond to the super states. For example, as illustrated in FIG. 4, only the chin and mouth regions are used to correspond to the super states, and there is no need to recognize other regions of the face. Whether the image contains face can be identified through training. As such, the amount of calculation can be reduced. The calculation process is similar as the above process, which is not repeated herein.

[0060] In at least one example, after using the embedded hidden Markov algorithm to

classify the test image into “face” image or “non-face” image, the image containing face is further classified, including identification of gender and age of the image. Although age and gender of a person cannot be accurately determined according to a state of the tongue, the state of the tongue of the person is related to its age and gender. For example, at different ages, the number of taste buds (papillary protrusions distributed on the tongue) is different, and it shows a decreasing trend. For children, there are about 10,000 taste buds. With age, cells are slowly aging. In old age, taste buds are only 20% of that in childhood. Furthermore, the younger, the tenderer the tongue and the redder the tongue; the older, the darker the tongue. The tongue of a female is usually smaller than that of a male. The test image can be classified according to age and gender, and then an image region containing tongue can be extracted from the image in each age-gender class (or category). The image in each class is more related to a tongue corresponding to the class. For example, if a tongue in an image belongs to the elderly, the image is classified into a class “older”. If the tongue in this image has characteristics of tongue that the elderly should have, such as a small number of taste buds (only a rough number) and dark colored tongue, it can be recognized more quickly. As such, the amount of calculation of the model can be reduced. It is to be noted, this requires training of an LNMF model corresponding to the age group in advance. That is, images in a training set are first classified according to age and gender and images in each class are labeled with “tongue contained” or “no tongue contained”. As such, a label regarding “age group-gender-tongue” can be generated. It is to be noted, train one LNMF model for each class.

[0061] The following are described in detail. A convolutional neural network (CNN) model is obtained, where the CNN model is trained and used to determine gender and age of the face. Assume that six classes are set according to age group and gender, for example, “0~20-male”, “20~40-male”, “40~70-male”, “0~20-female”, “20~40-female”, and “40~70-female”. The CNN is used for identification to classify the training images into the above six “age group-gender” classes.

[0062] Images in these six classes are labeled, and accordingly, a label corresponding to each image is a label regarding age group, gender, and tongue.

[0063] Train one LNMF model for each of these six classes, to obtain six LNMF models corresponding to these six classes. For example, an LNMF model corresponding to a tongue(s) of “0~20-male” is used to identify the tongue of “0~20-male”, and an LNMF model corresponding to a tongue(s) of “20~40-female” is used to identify the tongue of “20~40-female”.

[0064] The trained CNN model is used to identify gender and age of the test image. Similarly, the test image is classified according to the gender and age group, and classification of the test image is the same as that of the training image.

[0065] An image region containing tongue is extracted by using the trained LNMF model trained with age group, gender, and tongue.

[0066] The LNMF model is trained and is used to identify tongue images of corresponding “age group-gender” classes, the test image is also classified into a corresponding “age group-gender” class, and a tongue of this “age group-gender” class has some corresponding characteristics. As such, it is more beneficial for the LNMF model to perform identification. On the other hand, the test images are classified into multiple classes for simultaneous recognition, which speeds up efficiency of recognition. In addition, since the tongue images are labeled with age group and gender, it also beneficial to accurately and quickly perform subsequent classification on damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis.

[0067] FIG. 5 is a schematic diagram illustrating a hardware architecture of an electronic device according to implementations. In this example, the electronic device 2 is a device that can automatically perform numerical calculation and/or information processing according to pre-set or pre-stored instructions. For example, the electronic device 2 can be a smart phone, a tablet computer, a notebook computer, a desktop computer, a server, etc. As illustrated in FIG. 5, the electronic device 2 at least includes, but is not limited to, a memory 21, a processor 22, and a network interface 23 that can be communicatively coupled with each other through a system bus. The memory 21 includes at least one type of non-transitory computer readable storage medium. The readable storage medium includes flash memory, hard disk, multimedia card, memory, magnetic disk, optical disk, etc. In some examples, the memory 21 may be an internal storage unit of the electronic device 2, such as a hard disk or an internal storage of the electronic device 2. In other examples, the memory 21 may also be an external storage device of the electronic device 2. For example, a plug-in hard disk, smart media card (SMC), secure digital (SD) card, flash card, etc., equipped on the electronic device 2. In this example, the memory 21 is generally used to store an operating system and various application software installed in the electronic device 2, such as the program code for tongue image extraction.

[0068] In some examples, the processor 22 may be a central processing unit (CPU), controller, microcontroller, microprocessor, or other data processing chip. In this example, the processor 22 is used to run program codes or processing data stored in the memory 21, for

example, to run the program for tongue image extraction.

[0069] The network interface 23 may include a wireless network interface or a wired network interface. The network interface 23 is generally used to establish a communication connection between the electronic device 2 and other electronic devices.

[0070] In an example, the electronic device 2 may also include a display, where the display may also be called a display screen or a display unit.

[0071] The memory 21 containing a readable storage medium may include an operating system, a program for tongue image extraction 50, and the like. When the processor 22 executes the program for tongue image extraction 50 in the memory 21, the steps from S1 to S4 described above are implemented, which will not be repeated herein. In this example, the program for tongue image extraction 50 stored in the memory 21 may be divided into one or more program modules. The one or more program modules are stored in the memory 21 and can be executed by one or more processors (e.g., the processor 22 of this example) to achieve this application. For example, FIG. 6 is a schematic diagram illustrating program modules of the program for tongue image extraction. In this example, the program for tongue image extraction 50 can includes a matrix factorization module 501, a tongue features extracting module 502, and a tongue image segmenting module 503. The following description will introduce functions of the program modules.

[0072] The matrix factorization module 501 is configured to train with an LNMF algorithm, to obtain a feature base image of a different dimension. For example, 1000 tongue images (that is, images each containing tongue and reflecting shape and color of the tongue) are used as a set of training images, and the tongue images have been labeled in advance.

[0073] The LNMF is improvement of NMF. The LNMF algorithm refers to factorizing a matrix V corresponding to the training images into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH.

[0074] The matrix V has a dimension of n*m, where V= (V1, V2, ……, Vm). All non-negative gray-values of one image correspond to one column of the matrix V, and data in the matrix V is gray-values of the training images.

[0075] The non-negative feature matrix W has a dimension of n*r, and r columns are a base image.

[0076] The weight matrix H has a dimension of r*m, where each column of the weight matrix H is encoded and is in one-to-one correspondence with one tongue image in the matrix V. Therefore, one training image can be expressed as a linear combination of the base image.

[0077] In an example, NMF is a projection method based on local subspace. Features extracted by the NMF algorithm are global features, and no restriction is made on locality of a feature space. In order to enhance localization of principal components of the feature matrix W, LNMF emphasizes localization of basic feature components in original image decomposition. The formula of the LNMF algorithm is described above and is not repeated herein.

[0078] The tongue features extracting module 502 is configured to use an EHMM to recognize whether a test image contains face, and perform feature extraction on the test image when the test image contains face. The non-negative feature matrix W representing tongue features constitutes a non-negative subspace. The training images and the test image are respectively projected to the non-negative subspace obtained by the set of training images, to obtain feature coefficients. A nearest neighbor criterion is used to obtain similarities among the feature coefficients corresponding to the training images and the test image, to extract features of the test image. That is, when features have similarities higher than a set threshold, a feature base of the test image represents tongue. In this way, an image with the tongue features can be selected from the test image.

[0079] The tongue image segmenting module 503 is configured to project the test image to the non-negative subspace, where the projection is equivalent to transforming the test image into the non-negative subspace, and the test image is still an image which is composed of learned features. A non-feature region and a feature region have different labels, for example, the non-feature region is labeled with “0” and the feature region is labeled with “non-zero”. According to “0” and “non-zero”, an image region representing the tongue features can be segmented from the test image with a bounding box.

[0080] In at least one example, the program for tongue image extraction further includes a classification module 504. The classification module 504 is configured to use SVM classifiers to classify the features extracted from the test image, where the extracted features are input to k SVM classifiers for recognition and k is equal to the number of classes. In an example, the extracted features can be classified into “tongue” and “non-tongue”. In another example, the extracted features can also be classified according to pathological conditions of the tongue, so as to obtain an image of a tongue framed. In an example, as physical conditions of a person correspond to different features of tongue images, the classification can be performed according to the physical conditions. The physical conditions may include damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis. For the k SVM classifiers, a class with the highest score is used as a result for classification.

[0081] In at least one example, the program for tongue image extraction further includes a bounding box adjusting module 505. The bounding box adjusting module 505 is configured to use a linear regression model to adjust the position of the bounding box on the tongue image. For each class, for example, damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis, train one linear regression model, where input is features of the image in the bounding box and output is translation (left-right translation and up-down translation) values and a zoom value of the bounding box. The linear regression model is used to calculate the translation values and the zoom value of the bounding box and a loss function is used to constrain the position error of the bounding box, to adjust the bounding box to move to a suitable position.

[0082] In at least one example, the program for tongue image extraction further includes a binarizing module 506. The binarizing module 506 is configured to binarize the training images and the test image. The binarization refers to setting gray-values of pixels of the training images and the test image to be 0 or 255, that is, a process of presenting the entire image with clear black and white effect. For a color image (such as a RGB image), since a variety of colors are obtained through changes of three color channels of red (R), green (G), and blue (B) and their mutual superposition, a tongue region acquired from the color image may have more blank (missing) parts. However, a black-and-white image only has a single channel, and compared with the three color channels, the single channel is more beneficial for model optimization. As such, the tongue region can be acquired more accurately.

[0083] In at least one example, the program for tongue image extraction further includes a face recognition module 507. The face recognition module 507 is configured to first classify the test image using the EHMM. For example, the test image is classified into two classes: “face contained” and “no face contained”. As such, accuracy of recognition can be improved. The specific classification process is as follows.

[0084] Multiple feature points of the face are selected to form a feature sequence.

[0085] The test image is input into the EHMM for scanning the test image from top to bottom and from left to right through a sliding window. The EHMM includes a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction. Each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction. For the EHMM, the image is scanned from left to right and from top to bottom through a window of a fixed-size. The facial features can correspond to the super states from top to bottom and the embedded states

from left to right. For example, the sliding window first scans from left to right and each window obtains a group of eigenvectors through scanning, which is a type of feature extraction on a face region. After the sliding window obtains the eigenvectors, it moves to the right at a fixed distance to continue the feature extraction. When moving to the right side of the image, the sliding window switches to the next line to continue scanning from left to right. When the sliding window moves to the bottom right of the image, the whole scanning process ends and multiple groups of eigenvectors are obtained, where the multiple groups of eigenvectors form a sequence of observations.

[0086] A forward algorithm is used to derive a probability that the sequence of observations is similar as the feature sequence containing the multiple feature points of the face. Recognize that the test image contains face when the probability is greater than a determination threshold.

[0087] In at least one example, the EHMM is trained as follows.

[0088] 1) EHMM modeling: the EHMM can be defined as a ternary formula of $\lambda = (P_0, A_0, \Lambda)$, where $\Lambda = \Lambda^1, \Lambda^2, \cdots, \Lambda^{N_0}$, and basic elements of the EHMM include:

(1) an initial probability of a super state: $P_0 = \pi_{0,i}$, where $\pi_{0,i}$ is a probability of super state *i* at time=0, $1 \le i \le N_0$, and N_0 represents the number of super states;

(2) a transition matrix $A_0 = a_0, ij$, where a_0, ij is a probability that the transition matrix changes from super state *i* to super state *j*. In the EHMM from left to right, the only allowed transition is from a current state to a next state. Therefore, a probability from an original state to a previous state is 0;

(3) $\Lambda^k = (P_1^k A_1^k B^k)$ represents a parameter set of the k-th super state, where $1 \le k \le N_0$;

P_1^k is an initial probability distribution of an embedded state;

A_1^k a transition probability matrix of an embedded state;

B^k represents an observation probability matrix, $b_j^k(O_{t_0,t_1})$ represents a probability that embedded state j of super state k generates observation O_{t_0,t_1}, and two variables of O_{t_0,t_1} respectively correspond vertical and horizontal dimensions:

$$b_j^k(O_{t_0,t_1}) = \sum_{m=1}^{M_j^k} C_{jm}^k N(O_{t_0,t_1}, \mu_{jm}^k, U_{jm}^k),$$

M_j^k represents the number of mixed Gaussians;

C_{jm}^{k} is a mixing coefficient of the m-th mixing term of embedded state j of super state k;

$N(O_{t_0,t_1}, \mu_{jm}^{k}, U_{jm}^{k})$ is Gaussian density of mean vector μ_{jm}^{k} and covariance matrix $\bigcup_{jm}^{k}$.

[0089] 2) Image segmentation: the training image is uniformly divided, and a sequence of observations obtained from the image is evenly divided into N_0 longitudinal slices corresponding to longitudinal super states. Each longitudinal slice can be sliced into multiple embedded states from left to right.

[0090] 3) Parameter initialization: after segmentation, initial values of parameters of the model are obtained through an initialization probability and a transition probability of each state. Each state of the EHMM uses K-means clustering to calculate an observation probability. K is the number of Gaussian distributions in each state. For observation vectors extracted in all embedded states, a Gaussian mixture model can be used to explain an observation probability density function. The state initialization rule of each super state is as follows: an initialization probability of each first state of EHMM is specified as 1.0 and an initialization probability of other states is 0.

[0091] 4) Embedded Viterbi segmentation: after a first iteration, a dual embedded Viterbi algorithm is used to replace uniform segmentation, and through new segmentation and event frequency counting, a new set of initialization and transition probabilities are determined.

[0092] 5) Segmentation of K-means clustering: according to the segmentation result of step 4), K-means clustering is used to calculate an observation vector corresponding to a new state, and a new observation probability density function. In the next iteration, these values will be used as initial values for a new round of dual embedded Viterbi segmentation.

[0093] 6) Repeat steps 4) and 5) until change in the continuous iteration is less than a set convergence threshold.

[0094] In at least one example, the program for tongue image extraction further includes a re-classification module 508. The re-classification module 508 is configured to further classify the image containing face, including identification of gender and age of the image, after using the embedded hidden Markov algorithm to classify the test image into “face” image or “non-face” image.

[0095] The following are described in detail. A CNN model is obtained, where the CNN model is trained and used to determine gender and age of the face. Assume that six classes are

set according to age group and gender, for example, “0~20-male”, “20~40-male”, “40~70-male”, “0~20-female”, “20~40-female”, and “40~70-female”. The CNN is used for identification to classify the training images into the above six “age group-gender” classes.

[0096] Images in these six classes are labeled, and accordingly, a label corresponding to each image is “age group-gender-tongue”.

[0097] Train one LNMF model for each of these six classes, to obtain six LNMF models corresponding to these six classes. For example, an LNMF model corresponding to a tongue(s) of “0~20-male” is used to identify the tongue of “0~20-male”, and an LNMF model corresponding to a tongue(s) of “20~40-female” is used to identify the tongue of “20~40-female”.

[0098] The trained CNN model is used to identify gender and age group of the test image. Similarly, the test image is classified according to the gender and age, and classification of the test image is the same as that of the training image.

[0099] An image region containing tongue is extracted by using a corresponding LNMF model according to various classes.

[0100] The LNMF model is trained and is used to identify tongue images of corresponding “age group-gender” classes, the test image is also classified into a corresponding “age group-gender” class, and a tongue of this “age group-gender” class has some corresponding characteristics. As such, it is more beneficial for the LNMF model to perform identification. On the other hand, the test images are classified into multiple classes for simultaneous recognition, which speeds up efficiency of recognition. In addition, since the tongue images are labeled with age group and gender, it also beneficial to accurately and quickly perform subsequent classification on damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis.

[0101] This application further provides a device for tongue image extraction. The device for tongue image extraction includes a matrix factorization module 501, a tongue features extracting module 502, and a tongue image segmenting module 503.

[0102] The matrix factorization module 501 is configured to convert training images containing tongue into a matrix V, where all non-negative gray-values of one image correspond to one column of the matrix V, and train with an LNMF algorithm, to decompose the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH. The non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, where the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-

negative subspace, and the weight matrix H has a dimension of r*m and each column is encoded.

[0103] The tongue features extracting module 502 is configured to use an EHMM to recognize whether a test image contains face, project the training images and the test image to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, use a nearest neighbor criterion to obtain similarities among the feature coefficients corresponding to the training images and the test image, and extract features of the test image having similarities higher than a similarity threshold to be tongue features.

[0104] The tongue image segmenting module 503 is configured to identify, with different labels, a feature region with the tongue features and a non-feature region without the tongue features respectively and determine a minimum bounding box defining the feature region by reading the labels, such that the feature region representing the tongue features is segmented from the test image.

[0105] In at least one example, the device for tongue image extraction further includes a classification module 504. The classification module 504 is configured to use SVM classifiers to classify the features extracted from the test image, where the extracted features are sent to k SVM classifiers for recognition and k is equal to the number of classes.

[0106] In at least one example, the device for tongue image extraction further includes a bounding box adjusting module 505. The bounding box adjusting module 505 is configured to use a linear regression model to adjust the position of the bounding box on the tongue image. For each class, for example, damp heat, yin deficiency, normal, heat-rich, qi and blood barrier, and blood stasis, train one linear regression model, where input is features of the image in the bounding box and output is translation (left-right translation and up-down translation) values and a zoom value of the bounding box. The linear regression model is used to calculate the translation values and the zoom value of the bounding box and a loss function is used to constrain the position error of the bounding box, to adjust the bounding box to move to a suitable position.

[0107] In at least one example, the device for tongue image extraction further includes a binarizing module 506. The binarizing module 506 is configured to binarize the training images and the test image.

[0108] In at least one example, the device for tongue image extraction further includes a face recognition module 507. The face recognition module 507 is configured to first classify the test image using the EHMM. For example, the test image is classified into two classes: “face contained” and “no face contained”. Therefore, accuracy of recognition can be

improved.

[0109] This application further provides a non-transitory computer readable storage medium. The non-transitory computer readable storage medium can be any one or combination of hard disk, multimedia card, SD card, flash memory card, SMC, read only memory (ROM), erasable programmable read only memory (EPROM), portable compact disk read only memory (CD-ROM), USB memory, etc. The non-transitory computer readable storage medium includes a program for tongue image extraction, etc. When executed by the processor 22, the program for tongue image extraction 50 is configured to implement the following.

[0110] At S110, train with an LNMF algorithm, to obtain a feature base image of a different dimension. For example, 1000 tongue images (that is, images each containing tongue and reflecting shape and color of the tongue) are used as a set of training images, and the tongue images have been marked in advance. In an example, the tongue images can be compressed first, for example, compressed to 56*64 pixels, and the tongue images are de-averaged and normalized. The LNMF algorithm is used for training, to obtain the feature base image of a different dimension. The feature base image refers to a non-negative feature matrix W representing tongue features (i.e., features of tongue) and the non-negative feature matrix W constitutes a non-negative subspace.

[0111] The LNMF is improvement of NMF. The LNMF algorithm refers to factorizing a matrix V corresponding to the training images into a product of the non-negative feature matrix W and a weight matrix H, that is, V=WH.

[0112] The matrix V has a dimension of n*m, where V= (V1, V2, ……, Vm). All non-negative gray-values of one image correspond to one column of the matrix V, and data in the matrix V is gray-values of the training images.

[0113] The non-negative feature matrix W has a dimension of n*r, and r columns are a base image.

[0114] The weight matrix H has a dimension of r*m, where each column of the weight matrix H is encoded and corresponds to one tongue image in the matrix V. Therefore, one training image can be expressed as a linear combination of the base image.

[0115] At S120, an EHMM is used to recognize whether a test image contains face, and feature extraction is performed on the test image when the test image contains face. The non-negative feature matrix W representing tongue features constitutes a non-negative subspace. The training images and the test image are respectively projected to the non-negative

subspace obtained by the set of training images, to obtain feature coefficients. A nearest neighbor criterion is used to obtain similarities among the feature coefficients corresponding to the training images and the test image, to extract features of the test image. That is, when features have similarities higher than a set threshold, a feature base of the test image represents tongue. In this way, an image with the tongue features can be selected from the test image.

[0116] At S130, the test image is projected to the non-negative subspace, where the projection is equivalent to transforming the test image into the non-negative subspace, and the test image is still an image which is composed of learned features. Identify, with different labels, a feature region with the tongue features and a non-feature region without the tongue features and determine a minimum bounding box defining the feature region by reading the labels, such that the feature region containing the tongue features is segmented from the test image. For example, the non-feature region is denoted by “0” and the feature region is denoted by “non-zero”. According to “0” and “non-zero”, an image region representing the tongue features can be segmented from the test image with a bounding box.

[0117] The implementations of the non-transitory computer readable storage medium of this application is substantially the same as the implementations of the method for tongue image extraction and the electronic device 2 described above, which will not be repeated herein.

[0118] The above descriptions are only some implementations of this application and are not used to limit this application. For those skilled in the art, various modifications and changes can be made. Any modification, equivalent substitution, improvement, etc. made within the spirit and principle of this application shall be included in the protection scope of this application.

What is claimed is:

1. A method for tongue image extraction, applicable to an electronic device, the method comprising:

S110, converting training images containing tongue into a matrix V, wherein all non-negative gray-values of one image correspond to one column of the matrix V, and training with a local non-negative matrix factorization (LNMF) algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH, wherein

the non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, wherein the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and

the weight matrix H has a dimension of r*m and each column is encoded;

S120, using an embedded hidden Markov model (EHMM) to recognize whether a test image contains face, projecting the training images and the test image to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, using a nearest neighbor criterion to obtain similarities among the feature coefficients corresponding to the training images and the test image, and extracting features of the test image representing tongue and having similarities higher than a similarity threshold to be the tongue features; and

S130, after projection, identifying, with different labels, a feature region with the tongue features and a non-feature region without the tongue features respectively, wherein a set of labels corresponds to boundary information of the feature region, extracting extreme values in up, down, left, and right directions of the boundary information, and determining a bounding box defining the feature region.

2. The method of claim 1, further comprising:

S150, using a linear regression model to calculate a translation value and a zoom value of the bounding box and using a loss function to constrain a position error of the bounding box, to adjust the bounding box to move to a suitable position.

3. The method of claim 1, further comprising:

before S110, binarizing the training images and the test image and setting gray-values of

pixels of the training images and the test image to be 0 or 255 according to a set gray-value threshold.

4. The method of claim 1, wherein using the EHMM to recognize that the test image contains face comprises:

selecting a plurality of feature points of the face to form a feature sequence;

inputting the test image into the EHMM to scan the test image from top to bottom and from left to right through a sliding window, to obtain a plurality of groups of eigenvectors, wherein the plurality of groups of eigenvectors form a sequence of observations; and

using a forward algorithm to derive a similar probability between the sequence of observations and the feature sequence containing the plurality of feature points of the face, and recognizing that the test image contains face when the similar probability is greater than a determination threshold, wherein

the EHMM comprises a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction, and

each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction.

5. The method of claim 1, wherein at S110, the training images are compressed, de-averaged, and normalized, and training with the LNMF algorithm to obtain the feature base image, wherein

subtract from each column element of the matrix V a mean value of the column element; and

normalize the training images according to a ratio of a difference between each column element of the matrix V and a minimum value in the column element to a difference between a maximum value and the minimum value in the column element.

6. The method of claim 4, wherein after recognizing that the test image contains face, classifying the test image into a corresponding age group-gender class and using a LNMF model

for respective age group-gender classes to extract tongue images in respective age group-gender classes comprise:

obtaining a convolutional neural network (CNN) model, wherein the CNN model is trained and used to determine gender and age group, and classifying the training images into respective age group-gender classes;

annotating the training images in respective age group-gender classes, wherein a label regarding age group, gender, and tongue is obtained for each training image;

training the LNMF model according to the label regarding age group, gender, and tongue to obtain a trained LNMF model;

using the trained CNN model to identify gender and age group of the test image and classifying the test image according to the gender and age group; and

extracting a tongue image by using the trained LNMF model trained with age group, gender, and tongue.

7. A device for tongue image extraction, comprising:

a matrix factorization module configured to convert training images containing tongue into a matrix V, wherein all non-negative gray-values of one image correspond to one column of the matrix V, and train with a local non-negative matrix factorization (LNMF) algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH, wherein

the non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, wherein the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and

the weight matrix H has a dimension of r*m and each column is encoded;

a tongue features extracting module configured to use an embedded hidden Markov model (EHMM) to recognize whether a test image contains face, project the training images and the test image to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, use a nearest neighbor criterion to obtain similarities among the feature coefficients corresponding to the training images and the test image, and extract features of the test image representing tongue and having similarities higher than a similarity threshold to be the

tongue features; and

a tongue image segmenting module configured to identify, with different labels, a feature region with the tongue features and a non-feature region without the tongue features respectively, wherein a set of labels corresponds to boundary information of the feature region, extract extreme values in up, down, left, and right directions of the boundary information, and determine a bounding box defining the feature region.

8. The device of claim 7, further comprising:

a bounding box adjusting module configured to use a linear regression model to calculate a translation value and a zoom value of the bounding box and use a loss function to constrain a position error of the bounding box, to adjust the bounding box to move to a suitable position.

9. The device of claim 7, further comprising a face recognition module configured to use the EHMM to recognize that the test image contains face, wherein the face recognition module configured to use the EHMM to recognize that the test image contains face is configured to:

select a plurality of feature points of the face to form a feature sequence;

input the test image into the EHMM to scan the test image from top to bottom and from left to right through a sliding window, to obtain a plurality of groups of eigenvectors, wherein the plurality of groups of eigenvectors form a sequence of observations; and

use a forward algorithm to derive a similar probability between the sequence of observations and the feature sequence containing the plurality of feature points of the face, and recognize that the test image contains face when the similar probability is greater than a determination threshold, wherein

the EHMM comprises a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction, and

each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction.

10. The device of claim 7, further comprising a binarizing module configured to binarize the

training images and the test image and set gray-values of pixels of the training images and the test image to be 0 or 255 according to a set gray-value threshold.

11. An electronic device, comprising a processor and a memory configured to store a program for tongue image extraction, which when executed by the processor, is configured to implement:

S110, converting training images containing tongue into a matrix V, wherein all non-negative gray-values of one image correspond to one column of the matrix V, and training with a local non-negative matrix factorization (LNMF) algorithm, to factorize the matrix V into a product of a non-negative feature matrix W and a weight matrix H, that is, V=WH, wherein

the non-negative feature matrix W has a dimension of n*r, and r columns are a feature base image, wherein the feature base image refers to the non-negative feature matrix W representing tongue features and the non-negative feature matrix W constitutes a non-negative subspace, and

the weight matrix H has a dimension of r*m and each column is encoded;

S120, using an embedded hidden Markov model (EHMM) to recognize whether a test image contains face, projecting the training images and the test image to the non-negative subspace respectively to obtain feature coefficients when the test image contains face, using a nearest neighbor criterion to obtain similarities among the feature coefficients corresponding to the training images and the test image, and extracting features of the test image representing tongue and having similarities higher than a similarity threshold to be the tongue features; and

S130, after projection, identifying, with different labels, a feature region with the tongue features and a non-feature region without the tongue features respectively, wherein a set of labels corresponds to boundary information of the feature region, extracting extreme values in up, down, left, and right directions of the boundary information, and determining a bounding box defining the feature region.

12. The electronic device of claim 11, wherein the program for tongue image extraction, when executed by the processor, is further configured to implement:

before S110, binarizing the training images and the test image and setting gray-values of pixels of the training images and the test image to be 0 or 255 according to a set gray-value

threshold.

13. The electronic device of claim 11, wherein the program for tongue image extraction, when executed by the processor, is further configured to implement:

S150, using a linear regression model to calculate a translation value and a zoom value of the bounding box and using a loss function to constrain a position error of the bounding box, to adjust the bounding box to move to a suitable position.

14. The electronic device of claim 12, wherein the program for tongue image extraction, when executed by the processor, is further configured to implement: using the EHMM to recognize that the test image contains face, which comprises:

selecting a plurality of feature points of the face to form a feature sequence;

inputting the test image into the EHMM to scan the test image from top to bottom and from left to right through a sliding window, to obtain a plurality of groups of eigenvectors, wherein the plurality of groups of eigenvectors form a sequence of observations; and

using a forward algorithm to derive a similar probability between the sequence of observations and the feature sequence containing the plurality of feature points of the face, and recognizing that the test image contains face when the similar probability is greater than a determination threshold, wherein

the EHMM comprises a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction, and

each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction.

15. The electronic device of claim 12, wherein the program for tongue image extraction, when executed by the processor, is further configured to implement:

at S110, the training images are compressed, de-averaged, and normalized, and training with the LNMF algorithm to obtain the feature base image, wherein

subtract from each column element of the matrix V a mean value of the column

element; and

normalize the training images according to a ratio of a difference between each column element of the matrix V and a minimum value in the column element to a difference between a maximum value and the minimum value in the column element.

16. The electronic device of claim 12, wherein the program for tongue image extraction, when executed by the processor, is further configured to implement: after recognizing that the test image contains face, classifying the test image into a corresponding age group-gender class and using a LNMF model for respective age group-gender classes to extract tongue images in respective age group-gender classes, which comprise:

obtaining a convolutional neural network (CNN) model, wherein the CNN model is trained and used to determine gender and age group, and classifying the training images into respective age group-gender classes;

annotating the training images in respective age group-gender classes, wherein a label regarding age group, gender, and tongue is obtained for each training image;

training the LNMF model according to the label regarding age group, gender, and tongue to obtain a trained LNMF model;

using the trained CNN model to identify gender and age group of the test image and classifying the test image according to the gender and age group; and

extracting a tongue image by using the trained LNMF model trained with age group, gender, and tongue.

17. A non-transitory computer readable storage medium storing a computer program, the computer program comprising program instructions, which when executed by a processor, are configured to implement the method for tongue image extraction of claim 1.

18. The non-transitory computer readable storage medium of claim 17, wherein the program instructions, when executed by the processor, are further configured to implement: using the EHMM to recognize that the test image contains face, which comprises:

selecting a plurality of feature points of the face to form a feature sequence;

inputting the test image into the EHMM to scan the test image from top to bottom and

from left to right through a sliding window, to obtain a plurality of groups of eigenvectors, wherein the plurality of groups of eigenvectors form a sequence of observations; and

using a forward algorithm to derive a similar probability between the sequence of observations and the feature sequence containing the plurality of feature points of the face, and recognizing that the test image contains face when the similar probability is greater than a determination threshold, wherein

the EHMM comprises a set of super states, and the number of super states in the set of super states is the same as the number of slice images of the face in a vertical direction, and

each super state packages a set of embedded states, and the number of embedded states in the set of embedded states is the same as the number of slice images of the face in a horizontal direction.

19. The non-transitory computer readable storage medium of claim 17, wherein the program instructions, when executed by the processor, are further configured to implement:

at S110, the training images are compressed, de-averaged, and normalized, and training with the LNMF algorithm to obtain the feature base image, wherein

subtract from each column element of the matrix V a mean value of the column element; and

normalize the training images according to a ratio of a difference between each column element of the matrix V and a minimum value in the column element to a difference between a maximum value and the minimum value in the column element.

20. The non-transitory computer readable storage medium of claim 17, wherein the program instructions, when executed by the processor, are further configured to implement: after recognizing that the test image contains face, classifying the test image into a corresponding age group-gender class and using a LNMF model for respective age group-gender classes to extract tongue images in respective age group-gender classes, which comprise:

obtaining a convolutional neural network (CNN) model, wherein the CNN model is trained and used to determine gender and age group, and classifying the training images into respective age group-gender classes;

annotating the training images in respective age group-gender classes, wherein a label regarding age group, gender, and tongue is obtained for each training image;

training the LNMF model according to the label regarding age group, gender, and tongue to obtain a trained LNMF model;

using the trained CNN model to identify gender and age group of the test image and classifying the test image according to the gender and age group; and

extracting a tongue image by using the trained LNMF model trained with age group, gender, and tongue.