Multi-Task Warped Gaussian Process for Personalized Age Estimation

Yu Zhang & Dit-Yan Yeung
Department of Computer Science and Engineering, Hong Kong University of Science and Technology
{zhangyu,dyyeung}@cse.ust.hk

Abstract

Automatic age estimation from facial images has aroused research interests in recent years due to its promising potential for some computer vision applications. Among the methods proposed to date, personalized age estimation methods generally outperform global age estimation methods by learning a separate age estimator for each person in the training data set. However, since typical age databases only contain very limited training data for each person, training a separate age estimator using only training data for that person runs a high risk of overfitting the data and hence the prediction performance is limited. In this paper, we propose a novel approach to age estimation by formulating the problem as a multi-task learning problem. Based on a variant of the Gaussian process (GP) called warped Gaussian process (WGP), we propose a multi-task extension called multi-task warped Gaussian process (MTWGP). Age estimation is formulated as a multi-task regression problem in which each learning task refers to estimation of the age function for each person. While MTWGP models common features shared by different tasks (persons), it also allows task-specific (person-specific) features to be learned automatically. Moreover, unlike previous age estimation methods which need to specify the form of the regression functions or determine many parameters in the functions using inefficient methods such as cross validation, the form of the regression functions in MTWGP is implicitly defined by the kernel function and all its model parameters can be learned from data automatically. We have conducted experiments on two publicly available age databases, FG-NET and MORPH. The experimental results are very promising in showing that MTWGP compares favorably with state-of-the-art age estimation methods.

1. Introduction

A facial image contains very rich information about a person, such as identity, gender, expression, ethnicity, pose and age. Because of this, many applications based on facial images have been studied in the research community, including eye detection, face detection, face verification, face recognition, gender classification, expression recognition, ethnic classification and pose estimation. Recently, age estimation has also aroused interests in the research community due to its potential applications in biometrics, security control, human-computer interaction and surveillance monitoring.

Since the age of a person is often approximated by an integer, age estimation may be treated as either a classification problem or a regression problem. In [17], age estimation was regarded as a multi-class classification problem using classification methods such as nearest neighbor classifier and artificial neural network. However, because of the common performance measures used for age estimation, e.g., mean absolute error, the age estimation problem with ages rounded to integers is more of a regression problem than a multi-class classification problem. For example, misclassifying the age of a newborn baby to 3 should not be treated the same as misclassifying it to 5.

In the pioneering work on age estimation [18], the problem was regarded as a regression problem instead. Using different strategies or additional information, the authors proposed four age estimators which can be grouped into two categories: *global age estimation* and *personalized age estimation*. Global age estimation is based on the assumption that the aging process is the same for all people and hence the same age estimator can be used for different people. On the other hand, personalized age estimation is based on the assumption that different people go through different aging processes and hence different (personalized) age estimators are needed. Their experimental results show that personalized age estimation generally outperforms global age estimation. Moreover, from their analysis, the age estimators for two persons are similar if they have similar facial appearance. Nevertheless, the aging functions used in [18], such as quadratic and cubic regression functions, are quite simple.

Since the aging process is very complicated, some global age estimators were proposed using more sophisticated regression methods, including kernel regression [32] and support vector regression [13]. Moreover, in [5, 11], the authors

978-1-4244-6985-7/10/$26.00 ©2010 IEEE

assumed that the images form an aging manifold and used some manifold learning methods such as conformal embedding analysis to discover the manifold structure in the data. After finding the manifold structure, some regression methods such as least squares regression and locally adjusted robust regression are used to make prediction. In [29], the authors proposed a metric learning method for regression problems which preserves the local manifold structure and applied the method to age estimation. Besides global age estimators, more sophisticated methods were also used for personalized age estimators. In [10, 9], a personalized age estimator was proposed by modeling the problem as a time series problem. In particular, the aging pattern of a person corresponds to a sequence of facial images arranged in the order of age and the training images form a space of aging patterns for all the persons in the training set. Since the aging patterns are of high dimensionality and the aging patterns are usually not complete due to sparsity of the training data, principal component analysis [14] is used to find a linear aging pattern subspace and the missing parts in the aging patterns are determined simultaneously using an EM-style algorithm. The methods in [10, 9] are linear, but a nonlinear extension was also proposed in [8]. Moreover, the aging patterns were extended in [7] from the matrix form to the tensor form so that tensor decomposition can be used for age estimation.

Besides research conducted based on the standard age estimation setting, Yan et al. [30, 31] considered uncertainty on the age by representing it as a range rather than just a single number and used a ranking or regression method to make estimation. Moreover, some feature extraction methods were also proposed for age estimation. Yan et al. [32] borrowed the idea of a universal background model in speech verification. Each image is first represented as an ensemble of coordinate patches, then a Gaussian mixture model (GMM) is trained on all coordinate patches from all images, and finally the image-specific distribution model is derived by adapting the mean vectors of the GMM. Guo et al. [13] proposed a biologically inspired feature extractor which is similar to some feature extractors for object recognition.

In addition to age estimation, many other problems related to age have also been studied. In [16], a hierarchical classification method was used to classify facial images. The effect of the aging process for face recognition was studied in [24] and that on face verification was studied in [22]. Ramanathan and Chellappa [23] used a craniofacial growth model to synthesize facial images. In [21], the gradient orientation pyramid was used to extract facial features which were then used with a support vector machine for face verification. Short term aging patterns were learned in [27] and then similar short term aging patterns were concatenated to form long aging patterns. Also, the influence of gender on age estimation was studied in [12].

From previous research, it is generally agreed that personalized age estimation is superior to global age estimation. Along the line of personalized age estimation, the main contribution of this paper is to formulate age estimation as a novel multi-task regression problem. Our method, called *multi-task warped Gaussian process* (MTWGP), is a multi-task extension of the warped Gaussian process (WGP) [26]. More specifically, age estimation for each person is treated as a task and each task is estimated using a WGP. While different tasks are similar in that they share some common information, MTWGP is more flexible because it does not require the regression functions to be identical. Unlike MTWGP, the method in [18] learns each task independently with no sharing between tasks. MTWGP is also different from those in [10, 9, 8, 7] in that MTWGP models each task as a regression problem but the other methods model each task as a time series problem. Moreover, unlike methods which need to specify the form of the regression functions [18] or determine many parameters in the functions using inefficient methods such as cross validation [32, 5, 11], the form of the regression functions in MTWGP is implicitly defined by the kernel function and all the model parameters of MTWGP can be learned from data automatically using efficient gradient methods.

In summary, the main contributions of this paper can be summarized as follows: (1) we are the first to formulate age estimation as a multi-task learning problem; (2) we propose a multi-task extension of WGP; and (3) MTWGP outperforms state-of-the-art age estimation methods based on extensive experiments conducted on the only two age databases that are publicly available to date.

2. Gaussian Process and Warped Gaussian Process

In this section, we briefly review the Gaussian process (GP) [25] and the warped Gaussian process (WGP) [26].

Suppose we are given a training set which consists of n labeled data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with the ith point $\mathbf{x}_i \in \mathbb{R}^d$ and its corresponding output $y_i \in \mathbb{R}$.

For the GP model, we define a latent variable f_i for each data point $\mathbf{x}_i$. The prior distribution of $\mathbf{f} = (f_1, \ldots, f_n)^T$ is defined as $\mathbf{f} \,|\, \mathbf{X} \sim \mathcal{N}(\mathbf{0}_n, \mathbf{K})$, where $\mathbf{0}_n$ denotes the $n \times 1$ zero vector, $\mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ denotes the multivariate (univariate) normal distribution with mean as $\mathbf{m}$ and covariance matrix (variance) as $\mathbf{\Sigma}$, $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and $\mathbf{K}$ denotes the kernel matrix defined on $\mathbf{X}$ using a kernel function $k_{\boldsymbol{\theta}}(\cdot, \cdot)$ parameterized by $\boldsymbol{\theta}$. The likelihood for each data point is defined based on the Gaussian noise model: $y_i \,|\, f_i \sim \mathcal{N}(f_i, \sigma^2)$, where σ specifies the noise level. Due to the independence property, we can express it in a vectorial form as $\mathbf{y} \,|\, \mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma^2 \mathbf{I}_n)$ where $\mathbf{y} = (y_1, \ldots, y_n)^T$ and $\mathbf{I}_n$ denotes the $n \times n$ identity matrix. Figure 1 below depicts the graphical model for GP.

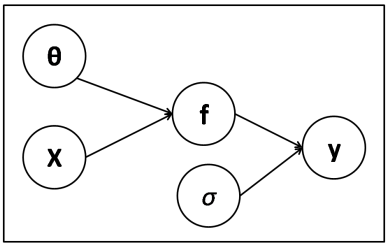

Figure 1. Graphical model for Gaussian process.

GP is a Bayesian kernel method. Compared with other kernel methods, especially the non-probabilistic ones, one advantage of GP is that its model parameters such as $\boldsymbol{\theta}$ and σ can be learned from data automatically without having to use model selection methods such as cross validation which requires training the model multiple times and hence incurs high computational cost. In Bayesian statistics, the marginal likelihood (also called model evidence) is usually used for learning the model parameters [6]. For GP in particular, since the marginal likelihood has a closed form, it is a good choice for model selection through parameter learning. It is easy to show that its marginal likelihood can be calculated as

$$p(\mathbf{y} \,|\, \mathbf{X}) = \int p(\mathbf{y} \,|\, \mathbf{f}) p(\mathbf{f} \,|\, \mathbf{X}) d\mathbf{f} = \mathcal{N}(\mathbf{0}_n, \mathbf{K} + \sigma^2 \mathbf{I}_n).$$

The negative log-likelihood of all data points $\mathbf{X}$, i.e., $-\log(p(\mathbf{y} \,|\, \mathbf{X}))$, can be expressed as follows after ignoring some constant terms

$$l = \frac{1}{2}\left[\mathbf{y}^T(\mathbf{K} + \sigma^2\mathbf{I}_n)^{-1}\mathbf{y} + \ln|\mathbf{K} + \sigma^2\mathbf{I}_n|\right],$$

where $\mathbf{A}^{-1}$ denotes the inverse of a square matrix $\mathbf{A}$ and $|\mathbf{A}|$ denotes its determinant. Here we use a gradient method to minimize l to estimate the optimal values of $\boldsymbol{\theta}$ and σ. Since each element of $\boldsymbol{\theta}$ and each element of σ are positive, we instead treat $\ln\theta_i$ and $\ln\sigma$ as variables where θ_i is the ith element of $\boldsymbol{\theta}$. The gradients of the negative log-likelihood with respect to each $\ln\theta_i$ and $\ln\sigma$ can be computed as:

$$\frac{\partial l}{\partial \ln\theta_i} = \frac{\partial l}{\partial\theta_i}\frac{\partial\theta_i}{\partial\ln\theta_i} = \frac{\theta_i}{2}\mathrm{tr}\left(\left[\tilde{\mathbf{K}}^{-1} - \tilde{\mathbf{K}}^{-1}\mathbf{y}\mathbf{y}^T\tilde{\mathbf{K}}^{-1}\right]\frac{\partial\mathbf{K}}{\partial\theta_i}\right)$$

$$\frac{\partial l}{\partial\ln\sigma} = \frac{\partial l}{\partial\sigma^2}\frac{\partial\sigma^2}{\partial\ln\sigma} = \sigma^2\mathrm{tr}\left(\tilde{\mathbf{K}}^{-1} - \tilde{\mathbf{K}}^{-1}\mathbf{y}\mathbf{y}^T\tilde{\mathbf{K}}^{-1}\right),$$

where tr$(\cdot)$ denotes the trace of a square matrix and $\tilde{\mathbf{K}} = \mathbf{K} + \sigma^2\mathbf{I}_n$.

After obtaining the optimal values of $\boldsymbol{\theta}$ and σ, we can make prediction for any unseen test data point. Given a test data point $\mathbf{x}_\star$, we need to determine the corresponding output $y_\star$. From the marginal likelihood, we can get

$$\begin{pmatrix} \mathbf{y} \\ y_\star \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}_{n+1}, \begin{pmatrix} \mathbf{K} + \sigma^2\mathbf{I}_n & \mathbf{k}_\star \\ \mathbf{k}_\star^T & k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma^2 \end{pmatrix}\right),$$

where $\mathbf{k}_\star = (k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_1), \ldots, k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_n))^T$. Then we obtain the predictive distribution $p(y_\star \,|\, x_\star, \mathbf{X}, \mathbf{y})$ as a Gaussian distribution with mean $m_\star$ and variance $\rho_\star^2$ as $m_\star = (\mathbf{k}_\star)^T(\mathbf{K} + \sigma^2\mathbf{I}_n)^{-1}\mathbf{y}$ and $\rho_\star^2 = k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma^2 - \mathbf{k}_\star^T(\mathbf{K} + \sigma^2\mathbf{I}_n)^{-1}\mathbf{k}_\star$. We can use $m_\star$ as the prediction for $\mathbf{x}_\star$.

Similar to GP, the latent variables $\mathbf{f}$ in WGP have the Gaussian prior and the likelihood is based on the Gaussian noise model. The difference, however, is that the likelihood in WGP is defined on $z_i = g(y_i)$, but not y_i, where the warped function $g(\cdot)$ is a monotonic real function parameterized by $\boldsymbol{\phi}$. The main idea behind WGP is that the original outputs $\{y_i\}$ do not satisfy the GP assumptions but their transformations $\{z_i\}$ do. The negative log-likelihood of WGP, after ignoring some constant terms, is given by

$$l = \frac{1}{2}\left[g(\mathbf{y})^T(\mathbf{K} + \sigma^2\mathbf{I}_n)^{-1}g(\mathbf{y}) + \ln|\mathbf{K} + \sigma^2\mathbf{I}_n|\right] - \sum_{i=1}^{n}\ln g'(y_i),$$

where $g(\mathbf{y}) = (z_1, \ldots, z_n)^T$ and $g'(y_i)$ denotes the derivative of function g at y_i. We then minimize the negative log-likelihood to obtain the optimal kernel parameters $\boldsymbol{\theta}$, noise level σ and parameters $\boldsymbol{\phi}$ in the function g simultaneously. When making prediction, suppose we are given a test data point $\mathbf{x}_\star$, we first calculate the predictive distribution for $z_\star$ as in GP, i.e., $\mathcal{N}(m_\star, \rho_\star^2)$, and then obtain the prediction for $\mathbf{x}_\star$ as $g^{-1}(m_\star)$ where $g^{-1}(\cdot)$ denotes the inverse function of $g(\cdot)$.[1]

3. Age Estimation as a Multi-Task Regression Problem

Previous research shows that personalized age estimation methods often outperform global age estimation methods [18, 10, 9, 8, 7]. The methods proposed in [18, 10] are the two main personalized age estimation methods in existence. The WAS method in [18] estimates the age function for each person in the training set independently. Since typical training sets for age estimation do not contain many images for each person, this method has a high risk of overfitting the training data and hence the prediction performance is limited. The AGES method in [10] needs to find the aging pattern subspace from the aging patterns defined on the training set. Also due to data sparsity in the training set, the aging patterns are incomplete and so AGES needs to estimate the missing parts relevant to age image synthesis, which is arguably an even harder problem than the original age estimation problem.

Multi-task learning [3, 2, 28] is a learning paradigm which seeks to improve the generalization performance of a learning task with the help of some other related tasks. This learning paradigm has been inspired by human learning activities in that people often apply the knowledge gained from previous learning tasks to help learn a new task. For example, a baby first learns to recognize human faces and later uses this knowledge to help it learn to recognize other

[1] When $g(\cdot)$ is not invertible, we can use numerical methods to find the preimage via the monotonic property of $g(\cdot)$.

objects. Multi-task learning can help to alleviate the small sample size problem which arises when each learning task contains only a very limited number of training data points. One widely adopted approach in multi-task learning is to learn a common data or model representation from multiple tasks [3, 19, 1] by leveraging the relationships between tasks. In this paper, we propose a novel approach to age estimation based on this idea from multi-task learning.

In age estimation, even though there exist (possibly substantial) differences between the aging processes of different individuals, some common patterns are still shared by them. For example, the face as a whole and all the facial parts will become bigger as one grows from a baby to an adult. Also, facial wrinkles generally increase as one gets older. As such, one may distinguish facial features of the aging process into two types. The first type is common to essentially all persons and the second type is person specific. From the perspective of multi-task learning, we want to use data from all tasks to learn the common features while task-specific (person-specific) features are learned separately for each task (person). Thus, age estimation can be formulated as a multi-task regression problem in which each learning task refers to estimation of the age function of each person. Suppose the age function of the ith person is approximated by the age estimator $h_i(x; \boldsymbol{\alpha}, \boldsymbol{\beta}_i)$, where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}_i$ are the common and person-specific model parameters corresponding to the two types of facial features. The learning problem thus corresponds to learning parameters of both types.

The only age estimation method related to ours is WAS [18] in which the age estimator for the ith person has the form $g_i(x; \boldsymbol{\gamma}_i)$, where $\boldsymbol{\gamma}_i$ models all the features of the ith person including features common to all persons and person-specific ones. Since $\boldsymbol{\gamma}_i$ includes all features, it is generally of high dimensionality. However, $\boldsymbol{\gamma}_i$ is estimated using only data for the ith person which is typically very limited (e.g., below 20 for the FG-NET database and about 3 for the MORPH database). As a result, it is difficult to estimate $\boldsymbol{\gamma}_i$ accurately. In our approach, since the common features are represented in $\boldsymbol{\alpha}$, $\boldsymbol{\beta}_i$ which captures the person-specific features for the ith person only has low dimensionality (equal to 1 in our method). Thus $\boldsymbol{\beta}_i$ can be estimated more accurately even though each task has very limited training data. Fig. 2 below summarizes the differences between WAS and our method. Moreover, since we model age estimation as a regression problem, we do not need to estimate the missing parts in the aging pattern and hence can overcome the limitations of AGES [10].

It is worth noting that the multi-task formulation may also be used for other variants of the age estimation problem. For example, if there is no identity information in the age database so that we do not know which image belongs to which person, we may generalize the concept of a task to other available information, such as gender or ethnicity. Taking gender information for example, we may assume that all male individuals share a common aging process and all female individuals share another common aging process. Thus one task corresponds to age estimation for male while another task for female. In this paper, we only consider the setting in which each learning task is for one person. Other variants will be investigated in our future research.

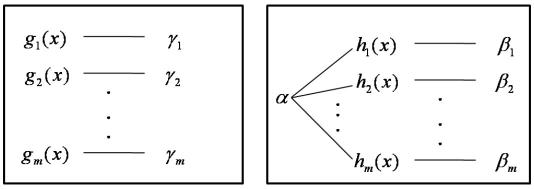

Figure 2. Modeling processes of WAS [18] (left) and our method (right). In WAS, the age estimator g_i for the ith person is parameterized by an independent parameter vector $\boldsymbol{\gamma}_i$. In our method, the age estimator h_i for the ith person is parameterized by a common parameter vector $\boldsymbol{\alpha}$ and a task-specific parameter vector $\boldsymbol{\beta}_i$.

4. Age Estimation using MTWGP

Suppose we are given a training set for age estimation consisting of m human subjects and so there are m tasks in total. The ith subject (task) has n_i data points $\{(\mathbf{x}_j^i, y_j^i)\}_{j=1}^{n_i}$ where $\mathbf{x}_j^i \in \mathbb{R}^d$ is the feature vector extracted from the image data and its output $y_j^i \in \mathbb{R}$ is the age of the person in the image. The total number of data points belonging to all tasks is $n = \sum_{i=1}^m n_i$.

4.1. Multi-Task Warped Gaussian Process

Our multi-task regression model for age estimation is called *multi-task warped Gaussian process* (MTWGP). We define a latent variable f_j^i for each data point $\mathbf{x}_j^i$. Similar to GP, the prior distribution of $\mathbf{f} = (f_1^1, \ldots, f_{n_1}^1, \ldots, f_1^m, \ldots, f_{n_m}^m)^T$ is given by

$$\mathbf{f} \,|\, \mathbf{X} \sim \mathcal{N}(\mathbf{0}_n, \mathbf{K}), \tag{1}$$

where $\mathbf{X}^i = (\mathbf{x}_1^i, \ldots, \mathbf{x}_{n_i}^i)$ denotes the data matrix for the ith task, $\mathbf{X} = (\mathbf{X}^1, \ldots, \mathbf{X}^m)$ denotes the total data matrix for all tasks, and $\mathbf{K}$ denotes the kernel matrix defined on $\mathbf{X}$ using a kernel function $k_{\boldsymbol{\theta}}(\cdot, \cdot)$ parameterized by $\boldsymbol{\theta}$.

Similar to WGP, the likelihood for each data point is defined on a latent variable z_j^i: $z_j^i \,|\, f_j^i \sim \mathcal{N}(f_j^i, \sigma_i^2)$, where $z_j^i = g(y_j^i)$ with the warped function $g(\cdot)$ parameterized by $\boldsymbol{\phi}$ and σ_i defines the noise level of the ith task. Note that this is different from both GP and WGP. In GP and WGP, the noise level is identical for all data points, but in MTWGP, the noise level is only the same for data points belonging to the same task. Similar to GP and WGP, we again can take advantage of the independence property to get the following vectorial form:

$$\mathbf{z} \,|\, \mathbf{f} \sim \mathcal{N}(\mathbf{f}, \mathbf{D}), \tag{2}$$

where $\mathbf{z} = (z_1^1, \ldots, z_{n_1}^1, \ldots, z_1^m, \ldots, z_{n_m}^m)^T$ and $\mathbf{D}$ denotes a diagonal matrix where a diagonal element is σ_i^2 if the corresponding data point belongs to the ith task.

Since we assume that different aging functions for different subjects share some similarities, we place a common prior on each σ_i to enforce all σ_i to be close to each other. Because σ_i are positive, we place a log-normal prior on them:

$$\sigma_i \sim \mathcal{LN}(\mu, \rho^2), \tag{3}$$

where $\mathcal{LN}(\mu, \rho^2)$ denotes the log-normal distribution with the probability density function as $p_t(\mu, \rho^2) = \frac{1}{t\rho\sqrt{2\pi}} \exp\Big\{ - \frac{(\ln t - \mu)^2}{2\rho^2} \Big\}$. This is equivalent to placing a common normal prior on $\ln \sigma_i$: $\ln \sigma_i \sim \mathcal{N}(\mu, \rho^2)$.

In summary, Eqs. (1), (2) and (3) are sufficient to define the whole model for MTWGP, demonstrating the simplicity (and beauty) of this model. Model parameters such as $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, μ and ρ are shared by all tasks corresponding to the common features, and $\{\sigma_i\}$ model the personal features of different subjects. The graphical model for MTWGP is depicted in Figure 3. In the next two subsections, we will discuss how to learn the model parameters and make prediction for new test data points.

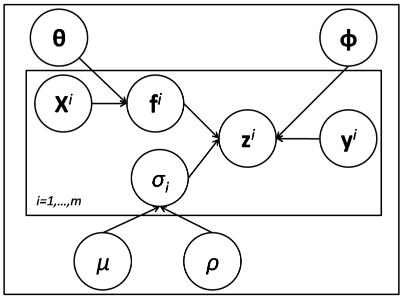

Figure 3. Graphical model for multi-task warped Gaussian process. $\mathbf{X}^i$ denotes the data matrix for the ith task, $\mathbf{f}^i = (f_1^i, \ldots, f_{n_i}^i)^T$, $\mathbf{y}^i = (y_1^i, \ldots, y_{n_i}^i)^T$ and $\mathbf{z}^i = (z_1^i, \ldots, z_{n_i}^i)^T$.

4.2. Model Parameter Learning

In MTWGP, the model parameters include $\{\sigma_i\}$, $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, μ and ρ. We maximize the likelihood to obtain the optimal values of these model parameters.

From the prior on $\mathbf{f}$ in Eq. (1) and the likelihood in Eq. (2), we can get

$$p(\mathbf{z} \,|\, \mathbf{X}) = \int p(\mathbf{f} \,|\, \mathbf{X}) p(\mathbf{z} \,|\, \mathbf{f}) d\mathbf{f} = \mathcal{N}(\mathbf{0}_n, \mathbf{K} + \mathbf{D}). \tag{4}$$

Recall that $z_j^i = g(y_j^i)$ and all σ_i share a common log-normal prior. So the likelihood for the training data can be calculated as $L = \mathcal{N}(g(\mathbf{y}) \,|\, \mathbf{0}_n, \mathbf{K} + \mathbf{D}) \prod_{i,j} g'(y_j^i) \prod_{i=1}^m \mathcal{LN}(\sigma_i \,|\, \mu, \rho^2)$, where $g(\mathbf{y}) = (g(y_1^1), \ldots, g(y_{n_m}^m))^T$ and the second term of L is the determinant of the Jacobian matrix when transforming the probability density function of $\mathbf{z}$ to that of $\mathbf{y}$.

For numerical stability, instead of maximizing the likelihood to obtain the optimal values of the model parameters, we solve an equivalent problem by minimizing the negative log-likelihood which is given as follows after ignoring some constant terms

$$L' = \frac{1}{2} \Big[g(\mathbf{y})^T (\mathbf{K} + \mathbf{D})^{-1} g(\mathbf{y}) + \ln |\mathbf{K} + \mathbf{D}| \Big] - \sum_{i=1}^m \sum_{j=1}^{n_i} \ln g'(y_j^i) + \sum_{i=1}^m \Big[\ln \sigma_i + \ln \rho + \frac{(\ln \sigma_i - \mu)^2}{2\rho^2} \Big].$$

Here we use gradient descent to minimize L'. The gradients of L' with respect to all model parameters are as follows:

$$\frac{\partial L'}{\partial \sigma_i} = \sigma_i \mathrm{tr}\Big([\tilde{\mathbf{K}}^{-1} - \tilde{\mathbf{K}}^{-1} g(\mathbf{y}) g(\mathbf{y})^T \tilde{\mathbf{K}}^{-1}] \mathbf{I}_n^i \Big) + \frac{\rho^2 + \ln \sigma_i - \mu}{\sigma_i \rho^2} \tag{5}$$

$$\frac{\partial L'}{\partial \theta_q} = \frac{1}{2} \mathrm{tr}\Big([\tilde{\mathbf{K}}^{-1} - \tilde{\mathbf{K}}^{-1} g(\mathbf{y}) g(\mathbf{y})^T \tilde{\mathbf{K}}^{-1}] \frac{\partial \mathbf{K}}{\partial \theta_q} \Big) \tag{6}$$

$$\frac{\partial L'}{\partial \phi_q} = g(\mathbf{y})^T \tilde{\mathbf{K}}^{-1} \frac{\partial g(\mathbf{y})}{\partial \phi_q} - \sum_{i=1}^m \sum_{j=1}^{n_i} \frac{\partial \ln g'(y_j^i)}{\partial \phi_q} \tag{7}$$

$$\frac{\partial L'}{\partial \mu} = \frac{m\mu - \sum_{i=1}^m \ln \sigma_i}{\rho^2} \tag{8}$$

$$\frac{\partial L'}{\partial \rho} = \frac{m}{\rho} - \frac{\sum_{i=1}^m (\ln \sigma_i - \mu)^2}{\rho^3} \tag{9}$$

where $\tilde{\mathbf{K}} = \mathbf{K} + \mathbf{D}$, $\mathbf{I}_n^i$ denotes an $n \times n$ diagonal 0-1 matrix where a diagonal element is equal to 1 if the data point with the corresponding index belongs to the ith task (person), ϕ_q is the qth element of $\boldsymbol{\phi}$, and $\frac{\partial g(\mathbf{y})}{\partial \phi_q} = \big(\frac{\partial g(y_1^1)}{\partial \phi_q}, \ldots, \frac{\partial g(y_{n_m}^m)}{\partial \phi_q} \big)^T$. In our experiments, the warped function $g(\cdot)$ has the form $g(x) = a \ln(bx + c) + d$ where $a, b, c > 0$ are positive real numbers and d is a real number. So $\boldsymbol{\phi} = (a, b, c, d)^T$.

Since the number of model parameters is not small, we use an alternating method to optimize L'. More specifically, we first update $\{\sigma_i\}$ using gradient descent with the other model parameters fixed and then use gradient descent to update the other model parameters including $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, μ and ρ with $\{\sigma_i\}$ fixed. These two steps are repeated until convergence.

4.3. Prediction

Given a test data point $\mathbf{x}_\star$, we need to determine the corresponding output $y_\star$. We define $z_\star$ as $z_\star = g(y_\star)$. Since we do not know which task $\mathbf{x}_\star$ belongs to, we introduce a new noise level $\sigma_\star$ for $\mathbf{x}_\star$. From Eq. (4), we can get

$$\begin{pmatrix} \mathbf{z} \\ z_\star \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}_{n+1}, \begin{pmatrix} \mathbf{K} + \mathbf{D} & \mathbf{k}_\star \\ \mathbf{k}_\star^T & k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma_\star^2 \end{pmatrix} \right),$$

where $\mathbf{k}_\star = (k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_1^1), \ldots, k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_{n_m}^m))^T$. Then, the predictive distribution $p(z_\star \,|\, x_\star, \mathbf{X}, \mathbf{y})$ can be obtained as a Gaussian distribution with the following mean $m_\star$ and variance $\rho_\star^2$:

$$m_\star = (\mathbf{k}_\star)^T(\mathbf{K}+\mathbf{D})^{-1}\mathbf{z}$$
$$\rho_\star^2 = k_{\boldsymbol{\theta}}(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma_\star^2 - \mathbf{k}_\star^T(\mathbf{K}+\mathbf{D})^{-1}\mathbf{k}_\star.$$

We use $m_\star$ as the prediction for $z_\star$. Interestingly, note that $m_\star$ does not depend on $\sigma_\star$. So for a test data point, we do not need to know which task the test data point belongs to. This property is very promising because in many real applications some test data points may not even belong to any of the tasks in the training set. Utilizing the relationship between $\mathbf{y}$ and $\mathbf{z}$ that $\mathbf{z} = g(\mathbf{y})$ and $z_\star = g(y_\star)$, we can get the prediction for the output of $\mathbf{x}_\star$ as

$$y_\star = g^{-1}\Big((\mathbf{k}_\star)^T(\mathbf{K}+\mathbf{D})^{-1}g(\mathbf{y})\Big). \tag{10}$$

The algorithm for MTWGP is summarized in Table 1.

Table 1. Algorithm for Multi-Task Warped Gaussian Process

```
Learning Procedure
Input: training data $\mathbf{X}$, $\mathbf{y}$
Output: $\{\sigma_i\}$, $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\mu$ and $\rho$
Initialize model parameters $\{\sigma_i\}$, $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\mu$ and $\rho$;
Repeat
    Update $\{\sigma_i\}$ using gradient descent with the gradients in Eq. (5);
    Update $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\mu$ and $\rho$ using gradient descent with the gradients in Eqs. (6)–(9);
Until {the consecutive change is below a threshold}

Testing Procedure
Input: $\mathbf{x}_\star$, $\{\sigma_i\}$, $\boldsymbol{\theta}$, $\boldsymbol{\phi}$
Output: prediction $y_\star$
Calculate the prediction via Eq. (10)
```

4.4. Discussions

The method discussed above may also be used for other regression problems in computer vision, such as pose estimation. For pose estimation, estimating the pose of each person can be treated as a separate task and MTWGP can learn the mapping between facial features and pose angles in a way similar to that for age estimation.

In [26], the authors recommend using the hyperbolic tangent function $\tanh(\cdot)$ as the candidate for $g(\cdot)$, that is, $g(x) = a\tanh(b(x+c))$ where $a, b > 0$ and $\tanh(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$. Since $g(x)$ is bounded in the interval $(-a, a)$ due to the boundedness of $\tanh(\cdot)$ and levels off as x becomes too small or too large, two very large (or very small) inputs x_1 and x_2 may give function values that are essentially identical numerically. This property is undesirable for age estimation. Instead, we use the function $\ln(\cdot)$, which is unbounded, in our experiments. Using this function, we found that the performance of WGP is much better than that of GP.

The computational complexity of our model is $O(n^3)$ where n is the sample size. When n is small, our model is quite efficient requiring no speedup scheme. If n is large, we may use a sparse version of GP, such as informative vector machine [20], which selects data points based on information theory and use these selected points to learn the model parameters and make prediction. As a result, the complexity of our method can be reduced to $O(nl^2)$ where l is the number of selected data points. In summary, our method, possibly with the incorporation of a sparse version of GP, is very efficient to make the method practical.

5. Experiments

In this section, we report experimental results obtained on two publicly available age databases by comparing MTWGP with a number of related age estimation methods in the literature.

5.1. Experimental Setting

To our knowledge, the only two age databases that are publicly available to date are FG-NET[2] and MORPH [15]. Both databases are used in our comparative study. In the FG-NET database, there are totally 1002 facial images from 82 persons with 6-18 images per person labeled with the ground-truth ages. The ages vary over a wide range from 0 to 69. The sample images of one person in the database are shown in Figure 4. In the MORPH database, there are 1724 facial images from 515 persons with only about 3 labeled images per person. The ages also vary over a wide range from 15 to 68. Some sample images are shown in Figure 5.

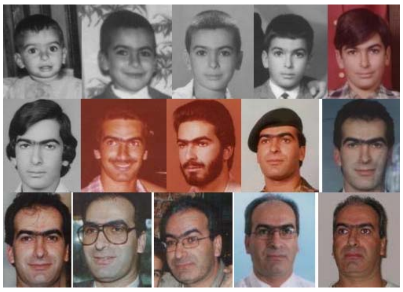

Figure 4. Sample images of one person in the FG-NET database.

Figure 5. Sample images of two persons in the MORPH database.

For fair comparison with methods reported in [18, 17, 10, 30, 31, 8, 11, 7, 29], we also use active appearance models (AAM) [4] for facial feature extraction in our experiments. According to [10, 9, 30, 31, 8, 11, 7, 29], we use two performance measures in our comparative study. The first one is the mean absolute error (MAE). Suppose there are t test images and the true and predicted ages of the kth image are denoted by a_k and $\tilde{a}_k$, respectively. The MAE is calculated as MAE $= \frac{1}{t}\sum_k |a_k - \tilde{a}_k|$, where $|\cdot|$ denotes the

[2] http://www.fgnet.rsunit.com/

absolute value of a scalar value. Another performance measure is referred to as the cumulative score. Of the t test images, suppose $t_{e\leq l}$ images have absolute prediction error no more than l (years). Then the cumulative score at error level l can be calculated as $\text{CumScore}(l) = t_{e\leq l}/t \times 100\%$. We only consider cumulative scores at error levels from 0 to 10 (years) since age estimation with an absolute error larger than 10 (i.e., a decade) is unacceptable for many real applications.

The methods compared are tested under the leave-one-person-out (LOPO) mode for the FG-NET database as in [9, 30, 31, 11, 8, 7, 29]. In other words, for each fold, all the images of one person are set aside as the test set and those of the others are used as the training set to simulate the situation in real applications. For the MORPH database, it has only about 3 images per person and is too few for training our MTWGP model. Thus, in our experiments, we adopt the the same setting as that in [9] by using the data in MORPH only for testing the methods trained on the FG-NET database.

5.2. Results on the FG-NET Database

Since the experimental settings are identical, we directly compare the results obtained by MTWGP with those reported in [9, 30, 31, 11, 8, 7, 29] obtained by other methods, which include WAS [18], AAS [17], AGES [10], RUN1 [30], RUN2 [31], LARR [11], KAGES [8], MSA [7], and mkNN [29]. We also include some other popular regression methods in our comparative study, including support vector regression (SVR)[3], GP[4] and WGP. Table 2 summarizes the results based on the MAE measure. We can see that MTWGP even outperforms other state-of-the-art methods for age estimation. Note that the performance of WGP is better than that of GP, showing that the proposed warped function $g(x)$ is very effective for this application. We also report in Figure 6 the results for different methods in terms of the cumulative scores at different error levels from 0 to 10, showing that MTWGP is the best at almost all levels.

5.3. Results on the MORPH Database

Again, due to the identical experimental setting used, we directly compare our results with those in [9] for WAS, AAS and AGES. Moreover, we also conduct experiments using other age estimation methods, including RUN1, LARR, mkNN, SVR, GP and WGP. The results in MAE and cumulative scores are recorded in Table 3 and Figure 7, respectively. We see again that MTWGP compares favorably with other state-of-the-art age estimation methods.

Table 2. Prediction errors (in MAE) of different age estimation methods on the FG-NET database.

Reference	Method	MAE
[9]	AAS	14.83
[9]	WAS	8.06
[9]	AGES	6.77
[8]	KAGES	6.18
[30]	RUN1	5.78
[7]	MSA	5.36
[31]	RUN2	5.33
[11]	LARR	5.07
[29]	mkNN	4.93
-	SVR	5.91
-	GP	5.39
-	WGP	4.95
-	MTWGP	**4.83**

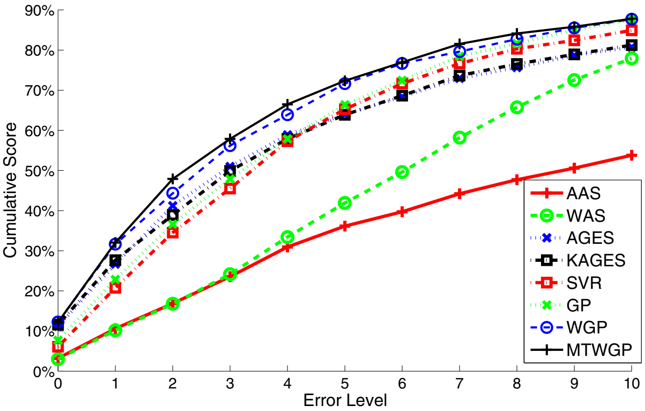

Figure 6. Cumulative scores (at error levels from 0 to 10 years) of different age estimation methods on the FG-NET database.

Table 3. Prediction errors (in MAE) of different age estimation methods on the MORPH database.

Reference	Method	MAE
[9]	AAS	20.93
[9]	WAS	9.32
[9]	AGES	8.83
-	RUN1	8.34
-	LARR	7.94
-	mkNN	10.31
-	SVR	7.20
-	GP	9.88
-	WGP	6.71
-	MTWGP	**6.28**

6. Conclusion

In this paper, we have proposed a novel formulation of the age estimation problem as a multi-task regression prob-

[3] http://www.csie.ntu.edu.tw/∼cjlin/libsvm/

[4] http://www.gaussianprocess.org/gpml/code/

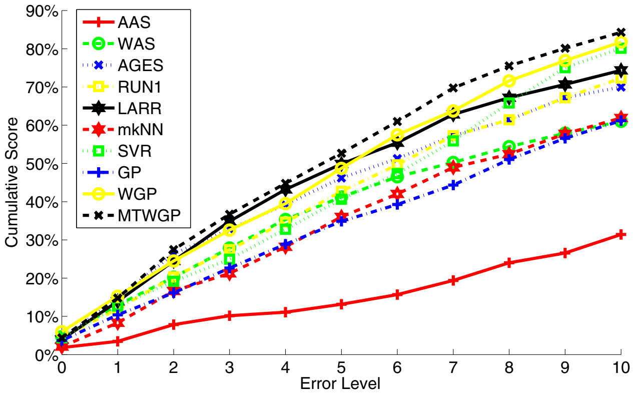

Figure 7. Cumulative scores (at error levels from 0 to 10 years) of different age estimation methods on the MORPH database.

lem. By proposing a multi-task extension of WGP, our MTWGP model may be viewed as a generalization of existing personalized age estimation methods by leveraging the similarities between the age functions of different individuals to overcome the data sparsity problem. Moreover, as a Bayesian model, MTWGP provides an efficient principled approach to model selection that is solved simultaneously when learning the age functions.

In our future research, we will apply MTWGP to other variants of the age estimation problem as well as other related regression problems in computer vision. Besides, we will also investigate multi-task extension of other regression methods, including SVR, regularized least square regression and so on.

Acknowledgment

We thank Xin Geng for providing the data and detailed experimental results of his previous studies on age estimation. This work is supported by General Research Fund 622209 from the Research Grants Council of Hong Kong.

References

[1] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 2008.

[2] J. Baxter. A Bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine Learning*, 1997.

[3] R. Caruana. Multitask learning. *Machine Learning*, 1997.

[4] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *TPAMI*, 2001.

[5] Y. Fu and T. S. Huang. Human age estimation with regression on discriminative aging manifold. *TMM*, 2008.

[6] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. 2003.

[7] X. Geng and K. Smith-Miles. Facial age estimation by multilinear subspace analysis. In *ICASSP*, 2009.

[8] X. Geng, K. Smith-Miles, and Z.-H. Zhou. Facial age estimation by nonlinear aging pattern subspace. In *ACMMM*, 2008.

[9] X. Geng, Z.-H. Zhou, and K. Smith-Miles. Automatic age estimation based on facial aging patterns. *TPAMI*, 2007.

[10] X. Geng, Z.-H. Zhou, Y. Zhang, G. Li, and H. Dai. Learning from facial aging patterns for automatic age estimation. In *ACMMM*, 2006.

[11] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang. Image-based human age estimation by manifold learning and locally adjusted robust regression. *TIP*, 2008.

[12] G. Guo, G. Mu, Y. Fu, C. Dyer, and T. Huang. A study on automatic age estimation using a large database. In *ICCV*, 2009.

[13] G. Guo, G. Mu, Y. Fu, and T. S. Huang. Human age estimation using bio-inspired features. In *CVPR*, 2009.

[14] I. T. Jolliffe. *Principal Component Analysis*. 2002.

[15] K. R. Jr. and T. Tesafaye. MORPH: A longitudinal image database of normal adult age-progression. In *AFGR*, 2006.

[16] Y. H. Kwon and N. V. Lobo. Age classification from facial images. *CVIU*, 1999.

[17] A. Lanitis, C. Draganova, and C. Christodoulou. Comparing different classifiers for automatic age estimation. *TSMC−Part B*, 2004.

[18] A. Lanitis, C. J. Taylor, and T. F. Cootes. Toward automatic simulation of aging effects on face images. *TPAMI*, 2002.

[19] N. D. Lawrence and J. C. Platt. Learning to learn with the informative vector machine. In *ICML*, 2004.

[20] N. D. Lawrence, M. Seeger, and R. Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In *NIPS 15*, 2003.

[21] H. Ling, S. Soatto, N. Ramanathan, and D. W. Jacobs. A study of face recognition as people age. In *ICCV*, 2007.

[22] N. Ramanathan and R. Chellappa. Face verification across age progression. *TIP*, 2006.

[23] N. Ramanathan and R. Chellappa. Modeling age progression in young faces. In *CVPR*, 2006.

[24] N. Ramanathan, R. Chellappa, and A. K. R. Chowdhury. Facial similarity across age, disguise, illumination and pose. In *ICIP*, 2004.

[25] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. 2006.

[26] E. Snelson, C. E. Rasmussen, and Z. Ghahramani. Warped Gaussian processes. In *NIPS 16*, 2004.

[27] J. Suo, X. Chen, S. Shan, and W. Gao. Learning long term face aging patterns from partially dense aging databases. In *ICCV*, 2009.

[28] S. Thrun. Is learning the n-th thing any easier than learning the first? In *NIPS 8*, 1996.

[29] B. Xiao, X. Yang, Y. Xu, and H. Zha. Learning distance metric for regression by semidefinite programming with application to human age estimation. In *ACMMM*, 2009.

[30] S. Yan, H. Wang, T. S. Huang, Q. Yang, and X. Tang. Ranking with uncertain labels. In *ICME*, 2007.

[31] S. Yan, H. Wang, X. Tang, and T. S. Huang. Learning auto-structured regressor from uncertain nonnegative labels. In *ICCV*, 2007.

[32] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang. Regression from patch-kernel. In *CVPR*, 2008.