

Skin Lesion Tracking using Structured Graphical Models

Hengameh Mirzaalian^{1,2}, Tim K Lee^{1,2,3}, and Ghassan Hamarneh¹

¹Medical Image Analysis Lab, Simon Fraser University, BC, Canada.

²Photomedicine Institute, Department of Dermatology and Skin Science, University of British Columbia and Vancouver Coastal Health Research Institute, BC, Canada.

³Cancer Control Research, BC Cancer Agency, BC, Canada.

Abstract

An automatic pigmented skin lesions tracking system, which is important for early skin cancer detection, is proposed in this work. The input to the system is a pair of skin back images of the same subject captured at different times. The output is the correspondence (matching) between the detected lesions and the identification of newly appearing and disappearing ones. First, a set of anatomical landmarks are detected using a pictorial structure algorithm. The lesions that are located within the polygon defined by the landmarks are identified and their anatomical spatial contexts are encoded by the landmarks. Then, these lesions are matched by labelling an association graph using a tensor based algorithm. A structured support vector machine is employed to learn all free parameters in the aforementioned steps. An adaptive learning approach (on-the-fly vs offline learning) is applied to set the parameters of the matching objective function using the estimated error of the detected landmarks. The effectiveness of the different steps in our framework is validated on 194 skin back images (97 pairs).

Keywords: Melanoma; Pigmented Skin Lesion; Anatomical Landmark; Feature (lesion and landmark) Detection; Lesion Tracking; Point Matching; Graph Matching; Graphical Models; Structured Support Vector Machines; Hyperparameter Learning; Error Prediction; Uncertainty Encoding.

1. Introduction

1.1. Background

Malignant melanoma (MM) is one of the common cancers among the white population [1]. It has been shown that the presence of a large number of pigmented skin lesions (PSLs) is an important risk factor for MM [2]. For early detection of MM, dermatologists advocate *total body photography* for high-risk patients. Regular examination and comparison of the skin using 2D digital pictures collected at different times can help identify newly-appearing, disappearing, and changing PSLs [3]. However, tracking PSLs in skin images is time consuming and error prone with large inter- and intra-rater variability. Recently, a few works have been proposed towards an automatic system for tracking PSLs [4, 5, 6], which mostly focused on the PSL matching task, where the positions of the PSLs and a set of anatomical landmarks (LNDs) are assumed to be known. In order to develop an end-to-end PSL tracking system, we need to detect the PSLs and LNDs automatically as well as perform PSL matching. In the remaining of the introduction section, we review the steps required for building such an automatic PSL tracking system.

[Ref]	Unary	Binary	Ternary	Optimization approach	Learning
[4]	✓	×	×	Dynamic Programming	×
[14, 15]	✓	✓	×	Taylor expansion and Gradient descent	×
[16]	✓	✓	✓	Taylor expansion and Gradient descent	✓
[17]	✓	✓	×	Dual decomposition	×
[18]	✓	✓	✓	Dual decomposition	×
[19, 20, 21, 22]	✓	×	×	Genetic algorithm	×
[23]	✓	×	×	Expectation-maximization	×
[24, 25, 26, 27, 28, 29, 30]	✓	✓	×	Expectation-maximization	×
[31]	✓	✓	×	Marginalization	×
[32]	✓	✓	✓	Marginalization	×
[33, 34, 6]	✓	✓	×	Spectral based	×
[35, 36]	✓	✓	×	Spectral based	✓
[37, 38, 5]	✓	✓	✓	Spectral based	×
Proposed	✓	✓	✓	Spectral based	✓

Table 1: Comparison between different point matching methods in terms of the order of the energy terms in the cost function (unary, binary, and ternary), the optimization approach, and the hyperparameter learning.

LND Detection. There exist many techniques for automatic LND detection based on feature points [7, 8, 9, 10]. In general, these methods consider a set of appearance models of the LNDs and their geometric relations to regularize the final detected LNDs. To the best of our knowledge, no work has been done on automatic LND detection on skin-back images.

PSL Detection. A notable number of methods have been proposed for PSL segmentation on *dermoscopic* images, which show *close-ups* on lesions, but not for skin back images in total-body photography, which are two different problems. The following are the key existing works on PSL detection on wide-area skin images. Sang et al. [11] detected potential PSL candidates on skin images of human arms by using feature vectors of steerable filter responses as input to a support vector machine (SVM) classifier. Pierrard et al. [12] applied Laplacian of Gaussian filter to enhance PSLs and then defined a saliency measurement to select the potential enhanced structures on face skin images. The most closely related PSL detection method for our application (i.e. back images) is the one by Lee et al. [13], in which they, first, applied thresholding to the output of an image enhanced by an adaptive Gaussian kernel. Then, they detected potential PSL candidates in the thresholded binary image based on geometric feature (e.g. area and elongation).

PSL Matching. Finding the mapping between the PSLs in a pair of images can be formulated as a graph matching problem. There exists extensive research on point matching as a graph labeling problem. In general, these methods construct a matching cost function including unary, binary, and ternary terms to measure matching compatibilities between the single, pair-wise, and triplet-wise correspondences. A common basic constraint is that each vertex in one graph be mapped to at most one point in the other one, and vice versa. To search for a crisp solution or a fuzzy solution different optimization approaches have been applied to minimize the matching cost function, e.g. dual decomposition, spectral and tensor based formulation, successive projections to find marginalization-matrix, gradient descent on Taylor expansion, genetic algorithm, and expectation-maximization. In Table 1, we compare the state-of-the-art point matching algorithms in terms of the order of the cost function (single, pair-wise, and triplet-wise) and the optimization approach.

Despite extensive research on graph matching, only a few works specifically focus on PSL matching. Huang and Bergstresser [4] developed a PSL matching algorithm using a PSL-based Voronoi decomposition of the image space. However, the Voronoi decomposition changes dramatically when one or more PSLs

appear or disappear causing the matching to fail. To normalize PSL coordinates prior to PSL matching, we previously proposed a landmark-based non-rigid warping of the back images to a unit-square template [5, 6]. However, this approach suffers from the following weaknesses: it requires an accurate segmentation of the back silhouette; the warping is influenced by all landmarks equally, even when certain landmarks are clearly more stable than others; and undesired distortions may occur for subjects requiring large warps.

Another observation on the existing point matching algorithms is that only a few papers propose a solution for learning the optimal set of parameters for graph matching. Leordeanu et al. [35, 16] applied gradient descent to minimize the error of their non-convex spectral-based formulation. Caetano et al. [36] applied a max-margin structured estimation technique to learn the parameters of quadratic assignment relaxation problems.

1.2. Our contributions

We formulate all the aforementioned steps (LND detection, PSL localization, and PSL matching) as optimization problems. *Our first contribution* is parts-based graphical models [39] for detecting the LNDs. The detected LNDs serve two purposes: They restrict the search space to a polygon during lesion localization and encode the anatomical spatial context of lesions (Section 2.1). The latter encoding is a novel PSL descriptor that we leverage for PSL matching.

Our second contribution is to improve PSL-detection approaches by using a new set of Hessian based descriptors (Section 2.3). Applying a random forest (RF) classifier, we compute a likelihood map for the presence of PSLs. Then, only the pixels inside the aforementioned polygon having a large likelihood value and belonging to a large enough connected component, are used in the subsequent PSL matching (Section 2.3).

As *Our third contribution*, we devise a new landmark based PSL descriptor that encodes uncertainty in the automatically detected LNDs and does not rely on any warping (Section 2.4). This descriptor is reminiscent of shape context [27], with the key conceptual difference that we capture spatial context with respect to the anatomical landmarks without constructing any histogram.

Similar to many medical image analysis problems, our formulations require setting objective functions with hyperparameters. In the *fourth contribution*, we apply structured support vector machine (SSVM) [40] to learn the free parameters (Sections 2.2). Thus, the proposed system is called a structured skin lesion tracking system.

Because of the dependence of the PSL descriptors on the detected LNDs, in *our final contribution*, we propose an adaptive system that, first, predicts the error in the detected LNDs and, then leverages the measured error to adapt (on-the-fly vs offline learning) the PSL matching (Section 2.6).

In contrast to the earlier works on graph matching (Table 1), our PSL matching technique uses the three terms (unary, binary, ternary) together using a spectral based optimization algorithm. We learn the hyperparameters of our objective function using SSVM.

In Table 2, we compare existing works on automatic lesion tracking systems. In contrast to the previous works on skin lesion tracking, the three tasks of LND detection, PSL detection, and PSL matching are performed automatically. Further, our method is unique in including parameter learning and consideration of uncertainty and predicted-error.

We evaluate the effectiveness of the different steps in our framework on 194 skin back color images (97 pairs). The results are presented in Section 3, followed by concluding remarks in Section 4.

Method	Auto LND	Auto PSL	Auto Matching	Learning Params	Measuring Uncertainty	Predicting Error
[41, 4, 6, 42, 5]	×	×	✓	×	×	×
[13, 12, 11]	×	✓	×	×	×	×
[35, 16, 36]	×	×	✓	✓	×	×
[5]	×	×	✓	×	✓	×
Proposed	✓	✓	✓	✓	✓	✓

Table 2: Comparison between different methods in terms of automating different steps (LND detection, PSL detection, PSL matching), parameter learning, and consideration of uncertainty and predicted-error.

2. Method

A summary of our framework is presented in Figure 1. Given two 2D color skin images \mathcal{I} and \mathcal{I}' , the goal is to detect PSLs in these two images and match (correspond) them. Let \mathcal{V} represent the set of PSLs in \mathcal{I} (similarly for \mathcal{V}' in \mathcal{I}'). We define a matching matrix \mathcal{X} , such that if $\mathcal{V}_i \in \mathcal{V}$ matches (corresponds to) $\mathcal{V}'_j \in \mathcal{V}'$ then $\mathcal{X}_{ij} = 1$ (and 0 otherwise). Corresponding PSLs must have homologous positions with respect to anatomical landmarks (LNDs). As in [6], we use the left and right neck, shoulder, armpit, and hip points as our landmarks and denote their spatial coordinates by \mathcal{L}_i , $i = 1 \dots 8$, in \mathcal{I} (similarly for \mathcal{L}'_i in \mathcal{I}'). There are three main stages in our algorithm: (i) LND Detection, (ii) PSL Detection, and (iii) PSL Matching.

In short, we formulate all the LND detection, PSL detection and matching objectives as linear combinations of different terms as shown in Table 3. In Table 4, we provide a list of frequently used acronyms in the following sections.

Problem	Unknown	Objective function	Weights and terms (u: unary, b:binary, t: ternary)	Optimization method
LND detection	\mathcal{L} landmarks in image \mathcal{I} \mathcal{L}' landmarks in image \mathcal{I}'	$E(\mathcal{L}) = \mathbf{w}_L^T \phi_L(\mathcal{L}, \mathcal{I})$	$\mathbf{w}_L = \begin{bmatrix} \mathbf{w}_L^u \\ \mathbf{w}_L^b \\ \mathbf{w}_L^t \end{bmatrix}$ $\phi_L(\mathcal{L}) = \begin{bmatrix} \phi_L^u(\mathcal{L}) \\ \phi_L^b(\mathcal{L}) \end{bmatrix}$	Pictorial [39]
PSL detection	\mathcal{P} lesion locations in \mathcal{I} \mathcal{P}' lesion locations in \mathcal{I}'	$E(\mathcal{P}) = \mathbf{w}_P^T \phi_P(\mathcal{P}, \mathcal{I})$	$\mathbf{w}_P = \begin{bmatrix} \mathbf{w}_P^u \\ \mathbf{w}_P^t \end{bmatrix}$ $\phi_P(\mathcal{P}) = \begin{bmatrix} \phi_P^u(\mathcal{P}) \end{bmatrix}$	Random forest [43]
PSL matching	\mathcal{X} matching matrix	$E(\mathcal{X}) = \mathbf{w}_M^T \phi_M(\mathcal{X}, \mathcal{P}, \mathcal{P}')$	$\mathbf{w}_M = \begin{bmatrix} \mathbf{w}_M^u \\ \mathbf{w}_M^b \\ \mathbf{w}_M^t \end{bmatrix}$ $\phi_M(\mathcal{X}) = \begin{bmatrix} \phi_M^u(\mathcal{X}) \\ \phi_M^b(\mathcal{X}) \\ \phi_M^t(\mathcal{X}) \end{bmatrix}$	Tensor based [37]

- $\{\phi^u, \phi^b, \phi^t\}$ correspond to the unary, binary, and ternary terms, respectively.
- $\{\phi^u, \mathbf{w}^u\} \in \mathbb{R}^u$, $\{\phi^b, \mathbf{w}^b\} \in \mathbb{R}^b$, $\{\phi^t, \mathbf{w}^t\} \in \mathbb{R}^t$.
- $\{\mathbb{R}^u, \mathbb{R}^b, \mathbb{R}^t\}$ dimensions depend on the unary, binary, and ternary descriptors in our formulations.

Table 3: General formulations of the LND detection, PSL detection and matching problems in our framework.

MM	Malignant melanoma	PSL	Pigmented skin lesions
LND	Landmark	RF	Random forest
SVM	Support vector machine	SSVM	Structured SVM

Table 4: Table of acronyms.

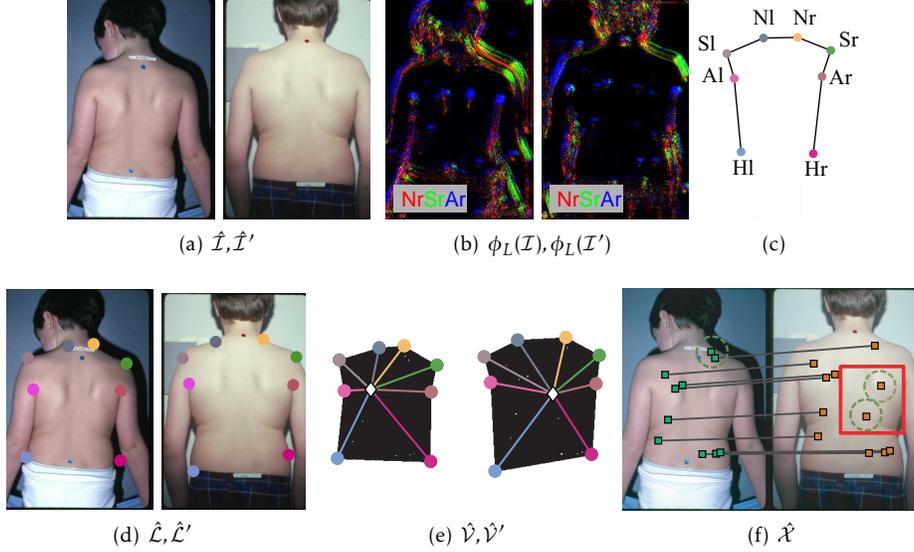


Figure 1: Summary of our framework. (a) Two skin back images of the same subject. (b) Computed likelihoods of the right neck, shoulder, and armpit LNDs encoded into the RGB channels (1). (c) The tree resulting from applying the minimum spanning tree algorithm computed in the training phase, which is used in the pictorial algorithm (Section 2). (d) Detected LNDs (1). (e) The overlaid colored line segments connecting the PSL to the landmarks visualize the approach used to compute the landmark context feature (14). The vertices of the black polygons are the LNDs in (d). (f) PSL correspondences (13) and two newly appearing PSLs are highlighted within the red box.

2.1. LND Detection

The goal here is to find the LNDs $\mathcal{L}_1 \dots \mathcal{L}_8 \in \mathcal{L}$ and $\mathcal{L}'_1 \dots \mathcal{L}'_8 \in \mathcal{L}'$, which we formulate as an optimization problem (similar optimization approach is applied to detect \mathcal{L}' in \mathcal{I}'):

$$\hat{\mathcal{L}}(\mathcal{I}) = \arg \min_{\mathcal{L}} \mathbf{w}_L^T \phi_L(\mathcal{L}, \mathcal{I}) \quad (1)$$

$$\mathbf{w}_L = [\mathbf{w}_L^1 \dots \mathbf{w}_L^8 \ \mathbf{w}_L^o]^T, \quad \phi_L(\mathcal{L}, \mathcal{I}) = [\phi_L^1(\mathcal{L}_1, \mathcal{I}) \dots \phi_L^8(\mathcal{L}_8, \mathcal{I}) \ \phi_L^o(\mathcal{L})]^T$$

where \mathcal{L}_i is a 2×1 vector representing 2D spatial coordinate of the i^{th} LND within the image domain, i.e. the search space for the coordinates of the i^{th} LND is the finite set of all image pixel coordinates; \mathbf{w}_L is a vector of scalars encoding weights of the different energy terms in ϕ ; $\phi_L^o(\mathcal{L}) = \sum_{ij} \phi_L^o(\mathcal{L}_i, \mathcal{L}_j)$ is the sum of the mahalanobis distance between all landmark pairs:

$$\phi^o(\mathcal{L}_i, \mathcal{L}_j) = (\mathcal{L}_{ij} - \bar{\mathcal{L}}_{ij})^T \mathcal{C}_{ij}^{-1} (\mathcal{L}_{ij} - \bar{\mathcal{L}}_{ij}) \quad (2)$$

where $\bar{\mathcal{L}}_{ij}$ and \mathcal{C}_{ij} are the mean and covariance matrix of the edges (vectors) $\mathcal{L}_{ij} = \mathcal{L}_i - \mathcal{L}_j$ (connecting \mathcal{L}_i to \mathcal{L}_j) in the training dataset:

$$\bar{\mathcal{L}}_{ij} = \frac{\sum_{n=1}^N \mathcal{L}_{ij}^n}{N}, \quad \mathcal{C}_{ij} = \frac{\sum_{n=1}^N [\mathcal{L}_{ij}^n - \bar{\mathcal{L}}_{ij}][\mathcal{L}_{ij}^n - \bar{\mathcal{L}}_{ij}]^T}{N-1} \quad (3)$$

where \mathcal{L}^k represents ground truth landmark information of the k^{th} image. This pairwise regularization term encourages the vector \mathcal{L}_{ij} to conform to a learnt graphical model.

$\phi_L^i(\mathcal{L}_i, \mathcal{I})$ in (1), $i = 1 \dots 8$, are the data (unary) terms that penalize locating the i^{th} LND at \mathcal{L}_i , each is captured using: local binary pattern-based features [44] of the RGB and HSV channels (6 features), xy spatial coordinates of the pixels (2 features), and Frangi et al. [45] filtered response (2 scales). Considering a 5×5 local window, the final feature vector to compute the data terms ϕ_L^i is of size 250 ($=25 \times [6+2+2]$). We extract the aforementioned features from 200 windows (per image in the training dataset) centered around the position of the landmarks as true positive samples and a set of randomly selected windows not near the position of the landmarks as true negative samples. These positive and negative samples are used to train a random forest classifier. Examples of the measured ϕ_L^i , $i = 1 \dots 3$, are shown in Figure 1(b).

We apply the pictorial structures algorithm [39] to globally optimize the final LND detection cost function (1). The algorithm in [39] is a recursive algorithm that requires computing the regularization term over an acyclic graph (tree) connecting the LNDs. To find the tree structure, we compute the minimum spanning tree (MST) over a complete graph with the 8 LNDs as vertices and with edge weights set as the covariance values of the edges [39]. Example of the computed MST tree is shown in Figure 1(c).

2.2. Structured Hyperparameter Learning

The LND optimization in (1) requires the knowledge of \mathbf{w}_L . We apply SSVM [40] to estimate \mathbf{w}_L by minimizing a LND loss function Δ_L computed over N training images:

$$\mathbf{w}_L = \arg \min_{\mathbf{w}} \frac{1}{N} \sum_{k=1}^N \Delta_L(\hat{\mathcal{L}}^k, \mathcal{L}^k) \quad (4)$$

$$\Delta_L(\hat{\mathcal{L}}^k, \mathcal{L}^k) = \sum_{i=1}^8 \|\hat{\mathcal{L}}_i^k - \mathcal{L}_i^k\|_2 \quad (5)$$

where $\hat{\mathcal{L}}_i^k$ is the predicted position of the i^{th} LND of the k^{th} image; and \mathcal{L}_i^k is the ground truth. Finding \mathbf{w}_L can be formulated as a max-margin problem:

$$\mathbf{w}_L = \min_{\mathbf{w}} \|\mathbf{w}\|_2^2 + C \sum_k \xi_k \quad (6)$$

$$\text{s.t. } \mathbf{w}^T \phi_L(\mathcal{L}^k, \mathcal{I}^k) \leq \mathbf{w}^T \phi_L(\mathcal{L}^k, \mathcal{I}^k) - \Delta_L(\mathcal{L}^k, \hat{\mathcal{L}}^k) + \xi_k, \forall \mathcal{L}, \forall k$$

where the slack variable ξ_k represents the upper bound of the risk for the k^{th} training sample; and C is the weight of the regularization term. The most violated constraint in (6) is captured by:

$$\arg \min_{\mathcal{L}} \mathbf{w}_L^T \phi_L(\mathcal{L}^k, \mathcal{I}^k) - \Delta_L(\mathcal{L}^k; \mathcal{L}^k) \quad (7)$$

Substituting (1) and (5) to the above equation, it can be seen that Δ_L^k in (7) can be encoded into the unary terms of (1):

$$\arg \min_{\mathcal{L}} \mathbf{w}_L^o \phi_L^o(\mathcal{L}) + \sum_{i=1}^8 \mathbf{w}_L^i \left(\phi_L^i(\mathcal{L}_i, \mathcal{I}) - \|\mathcal{L}_i^k - \hat{\mathcal{L}}_i^k\|_2 / \mathbf{w}_L^i \right) \quad (8)$$

Therefore, the pictorial algorithm [39] is also applicable to solve the above.

2.3. PSL Detection

The goal here is to find the PSL locations \mathcal{V}_i , $i = 1 \dots |\mathcal{V}|$, and $\mathcal{V}'_{i'}$, $i' = 1 \dots |\mathcal{V}'|$. Note that since PSLs may appear or disappear, $|\mathcal{V}|$ is not necessarily equal to $|\mathcal{V}'|$. Further, in our study, PSLs must be larger than 100 pixels $\approx 6 \text{ mm}^2$ to be clinically relevant [13]. Therefore, to prepare ground truth for the PSLs, we select large enough PSLs and annotate one point per PSL (please refer to Section 3 for more details on PSL ground truth preparation).

We set our PSL descriptors using features including: local binary pattern-based features [44] of the RGB and HSV channels (6 features), blobness and tubularness measurements resulting from singular value decomposition of the Hessian matrix [45] at two different scales (4 features). Considering 5×5 neighbouring pixels, the final feature vector to compute the data term of the PSLs is of size 250 ($= 25 \times [6+4]$). Given the extracted features $\phi_p(\mathcal{I}, p)$, we formulate the PSL detection problem as the optimization of the labelling of the pixels $p \in \mathcal{I}$ of the image, such that $\hat{\mathcal{P}}_p = 1$ if p is on the center of a PSL (0 otherwise):

$$\mathcal{P}_p(\phi_p(\mathcal{I}, p)) = \begin{cases} 1 & \text{if } p \text{ is on the center of a PSL} \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

Because of the high similarity between a pixel in the center of a PSL and its immediate neighboring points, we prepare fuzzy labels for the pixels of the image by fitting Gaussians centered around the positions of the PSLs \mathcal{P} :

$$\begin{aligned} \hat{\mathcal{P}}_p^* &= \exp\left(-\frac{\mathcal{D}(p)^2}{s^2}\right) \\ \mathcal{D}(p) &= \min_{\mathcal{P}_i \in \mathcal{P}} \|p - \mathcal{P}_i\|_2 \end{aligned} \quad (10)$$

where \mathcal{D} represents the distance transform of the pixels of the image with respect to the point set \mathcal{P} . To approximate $\hat{\mathcal{P}}_p^*$ by the extracted features $\phi_p(\mathcal{I}, p)$, we use Gradient boost regression trees [46]:

$$\hat{\mathcal{P}}_p = \sum_{k=1}^K \mathbf{w}_k h_k(\phi_p(\mathcal{I}, \mathcal{P})) \quad (11)$$

where \mathbf{w}_k and $h_k : \mathbb{R}^{|\phi_p|} \mapsto \mathbb{R}^1$ are a set of weights and weak learners, respectively, which are learnt iteratively to minimize a loss function defined between $\hat{\mathcal{P}}$ and $\hat{\mathcal{P}}^*$:

$$\begin{aligned} \{\mathbf{w}, h\} &= \arg \min_{\mathbf{w}, h} \frac{1}{N} \sum_{k=1}^N \Delta_p(\hat{\mathcal{P}}^k, \hat{\mathcal{P}}^k) \\ \Delta_p(\hat{\mathcal{P}}^k, \hat{\mathcal{P}}^k) &= \sum_{p \in \mathcal{I}^k} \|\hat{\mathcal{P}}_p^k - \hat{\mathcal{P}}_p^k\|_2 \end{aligned} \quad (12)$$

where $\hat{\mathcal{P}}_p^k$ is the predicted PSL score of the p^{th} pixel of the k^{th} image; and $\hat{\mathcal{P}}_p^k$ is the ground truth set as (10).

For a novel image, we seek an approximation of the center of the PSLs in term of the weighted sum of the outputs of the K weak learners as in (11). The search for \mathcal{P} is restricted by the polygon defined by the LNDs detected previously (Figure 1(d)). Given the computed likelihoods, we discretize them and choose detected binary regions with areas larger than 100 pixels to satisfy the PSL size constraint.

2.4. PSL Matching

The goal here is to find the matching matrix \mathcal{X} :

$$\hat{\mathcal{X}}(\mathcal{G}, \mathcal{G}') = \arg \min_{\mathcal{X}} \mathbf{w}_M^T \phi_M(\mathcal{X}, \mathcal{G}, \mathcal{G}') \quad (13)$$

$$\mathbf{w}_M = \begin{bmatrix} \mathbf{w}_M^u \\ \mathbf{w}_M^b \\ \mathbf{w}_M^t \end{bmatrix}, \quad \phi_M = \begin{bmatrix} \phi_M^u \\ \phi_M^b \\ \phi_M^t \end{bmatrix}, \quad \begin{aligned} \phi_M^u(\mathcal{X}, \mathcal{V}, \mathcal{V}') &= \sum_{ii'} \mathcal{X}_{ii'} \text{abs}(\mathcal{V}_i - \mathcal{V}'_{i'}), \quad \{\phi_M^u, \mathbf{w}_M^u, \mathcal{V}_i, \mathcal{V}'_{i'}\} \in \mathbb{R}^u \\ \phi_M^b(\mathcal{X}, \mathcal{E}, \mathcal{E}') &= \sum_{ii'} \sum_{jj'} \mathcal{X}_{ii'} \mathcal{X}_{jj'} \text{abs}(\mathcal{E}_{ij} - \mathcal{E}'_{i'j'}), \quad \{\phi_M^b, \mathbf{w}_M^b, \mathcal{E}_{ij}, \mathcal{E}'_{i'j'}\} \in \mathbb{R}^b, \\ \phi_M^t(\mathcal{X}, \mathcal{C}, \mathcal{C}') &= \sum_{ii'} \sum_{jj' kk'} \mathcal{X}_{ii'} \mathcal{X}_{jj'} \mathcal{X}_{kk'} \text{abs}(\mathcal{C}_{ijk} - \mathcal{C}'_{i'j'k'}), \quad \{\phi_M^t, \mathbf{w}_M^t, \mathcal{C}_{ijk}, \mathcal{C}'_{i'j'k'}\} \in \mathbb{R}^t \end{aligned}$$

where ϕ_M^u , ϕ_M^b , and ϕ_M^t measure matching compatibilities between the nodes, edges, and cliques; \mathcal{E}_{ij} and \mathcal{C}_{ijk} represent an edge and a clique connecting $\{\mathcal{V}_i, \mathcal{V}_j\}$ and $\{\mathcal{V}_i, \mathcal{V}_j, \mathcal{V}_k\}$, respectively.

The previous works on graph matching, which use the 2D coordinates of the nodes as node-descriptors, compute the node, edge, and clique compatibilities as a function of the differences between the nodes-coordinates (i.e. $\phi_M^1, \mathbf{w}_M^1 \in \mathbb{R}^{16}$), edge-lengths and edge-angles (i.e. $\phi_M^2, \mathbf{w}_M^2 \in \mathbb{R}^2$), triangle-area and triangle-angles (i.e. $\phi_M^3, \mathbf{w}_M^3 \in \mathbb{R}^4$) (Figure 2).

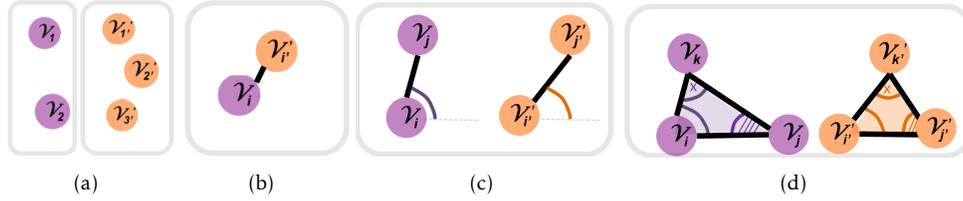


Figure 2: Schematic representation of two graphs \mathcal{G}_1 and \mathcal{G}_2 (a) and illustration examples of the single (b), pair-wise (c), and triplet-wise (d) compatibilities (please refer to Section 2.4 for further details).

In this work, we introduce a new graph descriptor called landmark context (LC), which encodes the relative position of the PSL with respect to the positions of all eight 2D LNDs. Given a pixel at position p in the image, its LC is a 16D vector computed by:

$$\mathcal{V}_p = \begin{pmatrix} (p - \mathcal{L}_1)/Z \\ \dots \\ (p - \mathcal{L}_8)/Z \end{pmatrix}, \quad Z = \sum_{i=1}^8 \|p - \mathcal{L}_i\|_2, \quad \mathcal{V} \in \mathbb{R}^{16}, \quad p \in \mathbb{R}^2. \quad (14)$$

Considering our proposed 16D descriptor, the dimensionality of the weights in (13) would become: $\mathbf{w}_M^1 \in \mathbb{R}^{16}$, $\mathbf{w}_M^2 \in \mathbb{R}^2$ (comparing the lengths and the angle of the edges (1D)), and $\mathbf{w}_M^3 \in \mathbb{R}^4$ (comparing the area (1D) and the angles of the triangles (3D)).

As earlier, we apply SSVM to learn the weights \mathbf{w}_M (13), which minimize a PSL matching loss function Δ_M over all the images:

$$\Delta_M(\hat{\mathcal{X}}, \mathcal{X}^*) = [\hat{\mathcal{X}} - \mathcal{X}^*]^T [\hat{\mathcal{X}} - \mathcal{X}^*]. \quad (15)$$

where $\hat{\mathcal{X}}$ and \mathcal{X}^* are the detected and the ground truth matching matrices, respectively; \odot represents an element-wise product between two matrices; and $\mathbf{1}$ is a matrix of ones of size equal to the size of $\hat{\mathcal{X}}$.

2.5. Adaptive Parameter Estimation for PSL Matching

There are different selection options for $\hat{\mathcal{X}}$ and $\hat{\mathcal{X}}$ in (15): (i) PSLs of the *most similar* subject(s) in the training stage (to find the closest pairs, we calculate the similarity between the graphs as a function of the difference between the PSLs, the sparsity of the PSLs, and the ratio of the eigenvalues of the covariance matrix of the coordinates) (ii) PSLs of *all* the images in the training data; (iii) PSLs and LNDs of the images in the training data; (iv) LNDs in the training data; and (v) LNDs of the test images (subject-specific).

The experimental results in Section 3 indicate that the last option (v), on-the-fly learning, is more useful than offline learning (options i-iv) only when LND detection is sufficiently accurate. In the on-the-fly learning option, the training data is created automatically from a novel image. Therefore, we learn to predict, for a novel image, the error in the detected positions of the LNDs (as explained next) then use the predicted error to automatically activate the on-the-fly learning of \mathbf{w}_M in (13).

2.6. LND Error Prediction

The goal here is to calculate, for a novel image, the error in the detected positions of the LNDs. Clearly, calculating the *exact* error requires the knowledge of a ground truth location. The idea here is to learn, from a training set, to *predict* the error using the following formulation:

$$\hat{\Delta}_L(\hat{\mathcal{L}}, \mathcal{I}) = \mathbf{w}_E^T \phi_E(\hat{\mathcal{L}}, \mathcal{I}) \quad (16)$$

$$\mathbf{w}_E = [\mathbf{w}_E^o \dots \mathbf{w}_E^{27}], \quad \phi_E(\hat{\mathcal{L}}, \mathcal{I}) = [\mathcal{S}(\hat{\mathcal{L}}) \quad \phi_L^o(\hat{\mathcal{L}}, \mathcal{I}) \quad \nabla \phi_L^o(\hat{\mathcal{L}}, \mathcal{I})^T \dots \quad \phi_L^8(\hat{\mathcal{L}}, \mathcal{I}) \quad \nabla \phi_L^8(\hat{\mathcal{L}}, \mathcal{I})^T]$$

where $\hat{\Delta}_L$ is the estimated error given the estimated locations $\hat{\mathcal{L}}$; $\mathcal{S}(\hat{\mathcal{L}})$ is an asymmetry feature that measures the difference between the edges connecting the left neck, shoulder, armpit, and hip points and their corresponding edges on the right side; $\phi_L^i(\hat{\mathcal{L}}, \mathcal{I})$ are the energy terms of the LND cost function in (1); ∇ is a gradient operator:

$$\nabla \phi_L^i(\hat{\mathcal{L}}_i, \mathcal{I}) = \left[\frac{\partial \phi_L^i(\hat{\mathcal{L}}_i, \mathcal{I})}{\partial x} \quad \frac{\partial \phi_L^i(\hat{\mathcal{L}}_i, \mathcal{I})}{\partial y} \right]^T$$

$$\nabla \phi_L^o(\hat{\mathcal{L}}) = \sum_{ij} \nabla \phi_L^o(\mathcal{L}_i, \mathcal{L}_j) = \sum_{ij} c_{ij}^{-1} (\hat{\mathcal{L}}_{ij} - \tilde{\mathcal{L}}_{ij})$$

We use these features to quantify the uncertainty in the probabilistic solution. It has been shown that the presence of uncertainty may indicate errors in the solution and this uncertainty can be quantified using the slope of each energy term (measured by their gradients) at the computed solution [47].

In a training stage, we calculate \mathbf{w}_E knowing $\Delta_L(\hat{\mathcal{L}}) = \Delta_L(\hat{\mathcal{L}}, \hat{\mathcal{L}})$ (6) and ϕ_E . In the testing stage, given a predicted error $\hat{\Delta}_L(\hat{\mathcal{L}})$ with low values, we perform the on-the-fly learning on the weights of the PSL matching algorithm, otherwise, we use the off-line learnt weights.

3. Results

We evaluated our method on 194 digital color images (97 pairs, i.e. two images per subject) of the human back with isotropic resolution of 0.25 mm/pixel [48].

We manually prepared ground truth for the locations of both the LNDs and PSLs in all the 97x2=194 images as well as their correspondences across each pair. We annotated one point (coordinate) per LND and PSL. We had eight LNDs in each image but the number of PSLs varied between 2 and 30 (9±8).

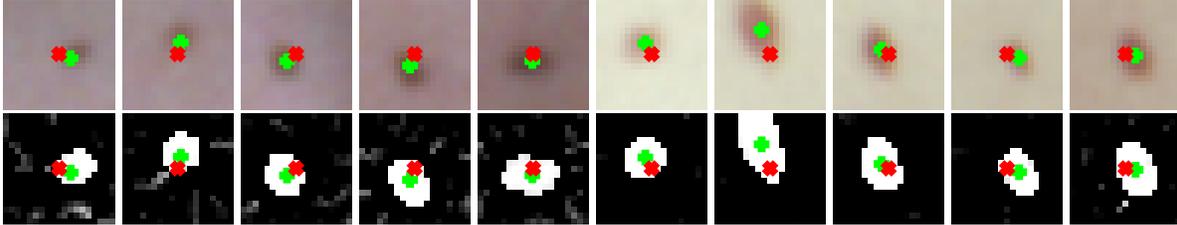


Figure 3: Post-processing to refine the manual PSL locations (red markers) by relocating them to the mass centers (green markers) of the blobness response patches around the annotated pixels (Section 3). The top row shows the image patches (centered at the positions of the red markers) and the bottom row shows the blobness responses.

Since the manual localization of the PSLs did not fall exactly in the centre of the PSL, we resort to an automatic post-processing step to refine the manual PSL coordinates such that it coincides with (or is relocated to) the “centre of mass” (CoM) of the PSL. In particular, the CoM is calculated for a blobness response (calculated as in [45]) patch of size 30×30 pixel centered around the annotated point (Figure 3).

Ten-fold cross-validation was performed for learning the hyperparameters of our objective functions for LND detection (1), PSL detection (11), and PSL matching (13). For each of the 10 validation runs, 20 images (10 pairs) were left for testing and the remaining images were used for training.

Note that the images in our dataset were taken from the patients in a similar posture and within a similar distance from the camera. Therefore, we had this underlying assumption that all the images are roughly in the same scale and orientation. This assumption mainly affects the LND detection step, which is solved by applying the pictorial algorithm [39]. This LND detection step, while not invariant to scale and rotation, is robust to small changes to those pose factors [39].

Samples of qualitative results of the detected LNDs, PSLs, and PSL correspondences are shown in Figure 4. In the following, we report the quantitative results of the different steps. Note that the results from [49, 13] were obtained by re-implementing the methods from the published papers.

3.1. LND Detection

Compared with our previous work reported in [5, 6], the current method detected LNDs fully automatically. We achieved an average of 10.76 ± 12.43 mm Euclidean distance error with respect to the ground truth over 1552 LNDs in 97 image pairs (8 LNDs/image). Note that an error of ~ 1 cm falls within the typical localization variability amongst human observers [50]. In Figure 5, we show the cumulative root mean square error (CRMSE) of the detected LNDs; CRMSE is a function that measures percentage of images having LND detection error less than a specific value, e.g.: given N images in the test-stage, CRMSE of the i^{th} LND, $i = 1 \dots 8$, with LND detection error $\leq z$, CRMSE is defined as:

$$\text{CRMSE}_i(z) = \frac{\sum_{k=1}^N \delta(\Delta_L(\hat{\mathcal{L}}_i^k, \mathcal{L}_i^k) < z)}{N} \times 100\%, \quad \delta(\cdot) = \begin{cases} 1 & \text{if } \Delta_L^k \leq z \\ 0 & \text{if } \Delta_L^k > z \end{cases} \quad (17)$$

In Figure 5, it can be seen that using SSVM to learn $\mathbf{w}_L \in \mathbb{R}^9$ produces better results than SSVM-learning of only two weights (one for the regularization and another equal weight for all data terms), i.e. $\mathbf{w}_L \in \mathbb{R}^2$ (Figure 5). Note that when equal weights are used for all (regularization and data) terms, we obtain unusable results, which are not included in Figure 5.

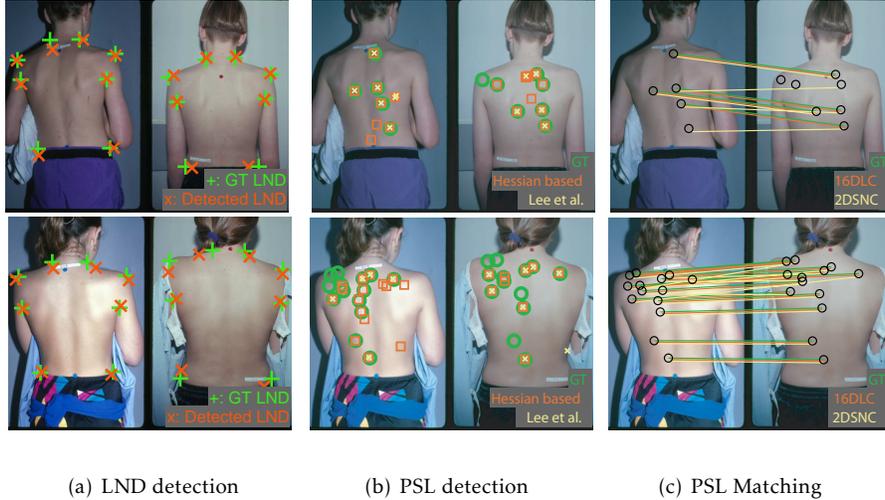


Figure 4: Sample qualitative results (see text for quantitative results): (a) Detected landmarks (red) using the graphical models (Section 2.1) compared with the ground truth (green). (b) Detected PSLs using the Hessian based features (red) and the method in [13] (yellow) compared with the ground truth (green). (c) PSL matching results using the 16DLC (red) and 2DSNC (yellow) descriptors compared with the ground truth (green) (Section 2.4).

3.2. PSL Detection

We validate the goodness of the detected PSLs in terms of true positive fraction (TPF) and positive predictive value (PPV) computed as the following:

$$\begin{aligned} \text{TPF} &= (\text{Number of true positives}) / (\text{Number of true positives} + \text{Number of false negatives}) \\ \text{PPV} &= (\text{Number of true positives}) / (\text{Number of true positives} + \text{Number of false positives}). \end{aligned} \quad (18)$$

Averaged over all the images, using the Hessian based features, we achieve TPF of 0.80 ± 0.06 and PPV of 0.87 ± 0.01 , which outperform state-of-the-art [13] (TPF= 0.68 ± 0.18 and PPV= 0.61 ± 0.01). Note that we consider a detected PSL as a true positive if there is an overlap of at least 5 mm^2 between the region of the detected PSL and a ground truth.

Note that [13] does not require training data (although it had hyperparameters that were empirically set by hand).

3.3. Structured Parameter Estimation

Figure 7(a) shows the PSL matching errors resulting from applying the different weight-learning options (Section 2.2). First, it can be seen that considering any of the SSVM-based learnt weights drops the matching error from about 0.3 to 0.1 (i.e. by 66.7%) compared with the non-learnt weights (e.g. equal weights). Second, the results indicate that, given a good enough estimation of the position of the LNDs, less than 20 mm (Figure 7(a)), learning the weights based on the automatically detected LNDs of the same subject (i.e. on-the-fly training) is more accurate and with less variability than using PSLs/LNDs from different subjects (off-line training).

In contrast to the earlier works on graph matching (Table 1), our PSL matching technique uses the three terms (unary, binary, ternary) together

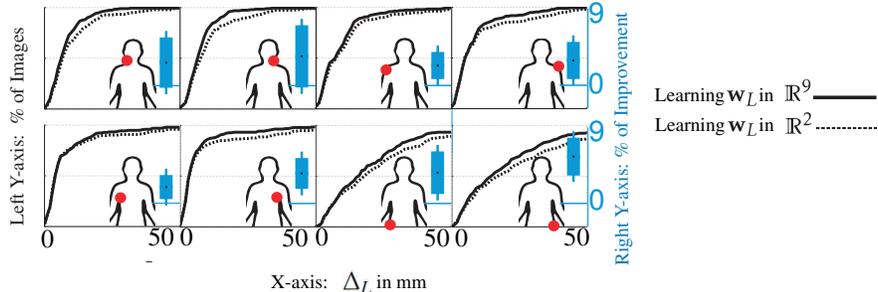


Figure 5: Cumulative RMSE of the eight detected LNDs when learning a single weight for all data terms, i.e. $\mathbf{w}_L \in \mathbb{R}^2$ for data and regularization (dashed curves) and when learning the weights for the regularization and all the 8 data terms, i.e. $\mathbf{w}_L \in \mathbb{R}^9$ (solid curves) using SSVM (1); the left y-axis represents percentage of images. The colored bars on the right-side of each figure represent percentage of the improvement (read of the right y-axis) resulting from the learnt parameters compared with non-learnt. X-axis is the LND detection error computed ((16)).

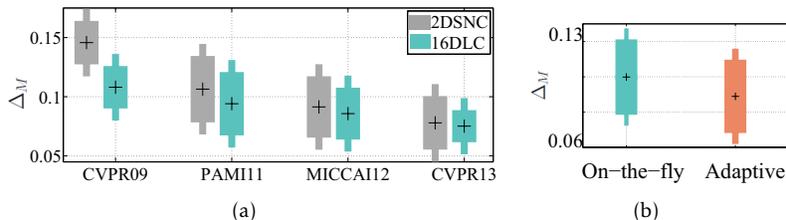


Figure 6: (a) PSL matching error when using 2DSNC (gray bars) and 16DLC (green bars). (b) PSL matching errors resulting from off-line (using LNDs), on-the-fly, and adaptive learning.

3.4. LND Context Effectiveness

Figure 6(a) shows the PSL matching errors Δ_M (15) of the following matching algorithms: CVPR09 [6], PAMI11 [37], MICCAI12 [5], and CVPR13 [49] using 2D spatially normalized coordinates (2DSNC) [6] and our proposed 16DLC (14). In summary, all these matching methods can be implemented by setting the objective matching functions including binary or ternary terms and applying different optimization approaches mentioned in Table 1. As it can be seen in Figure 6(a), applying 16DLC reduces the PSL matching error for all methods.

Note that in our previous works [5, 6, 50], we examined two other node descriptors: shape context [27] and Voronoi area [4]. The PSL matching performance resulting from applying these descriptors to our dataset was unacceptable for the following reasons. Our PSL matching problem can be considered as a sparse graph matching problem as the PSLs are not too dense. However, shape context relies on constructing a log-polar histogram from a dense set of points. Voronoi area descriptor, on the other hand, proved to be a weak descriptor for either sparse or dense graph matching because any points can have similar Voronoi areas in the presence of dense points. Further, in the presence of outliers (e.g the presence of newly appearing and disappearing PSLs in our work), two corresponding points could have totally different Voronoi areas.

3.5. LND Error Prediction

Figure 7(b) shows a scatter plot of the real and the predicted LND-error, which have a statistically significant ($p\text{-value} \leq 0.01$) correlation coefficient $\rho = 0.61$.

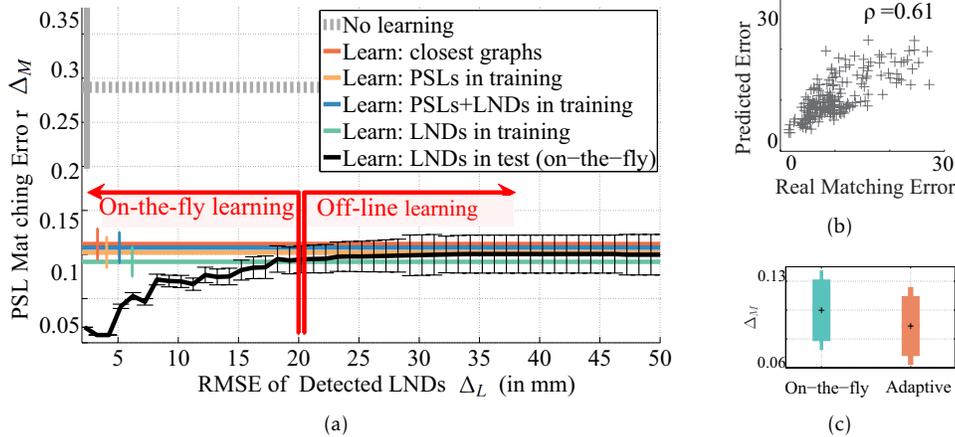


Figure 7: (a) The PSL Matching error over the RMSE of the detected LNDs without and with applying the different learning approaches to learn $\mathbf{w}_M \in \mathbb{R}^{37}$ weights in (13). (b) Scatter plot between the real and the predicted LND-errors. (c) PSL matching errors resulting from off-line (using LNDs), on-the-fly, and adaptive learning.

3.6. Adaptive Learning Effectiveness

Comparing the PSL matching errors resulting from on-the-fly and adaptive learning in Figure 7(c) demonstrates the advantage of the adaptive approach.

4. Conclusion

We presented an end-to-end automatic PSL-tracking system, which is important for early skin cancer detection. Our results demonstrate that the accuracy of the LND detection with SSVM-based parameter learning is similar to experts' annotations; the PSL detection accuracy outperforms the state-of-the-art methods; and the new PSL descriptor with the adaptive structured on-the-fly learning gives superior PSL matching results. During the development, we also made the following computational contributions: i) LND detection based on a structured graphical model, ii) polygon-constrained PSL detection, iii) a new graph descriptor we call *landmark context*; iv) structured subject-specific hyperparameter estimation for PSL matching; and v) LND error prediction for adaptive PSL matching. All of these contributions were validated and shown to improve performance.

As future work, we also consider jointly solving the above mentioned problems of PSL detection and matching. It is possible to consider the PSL detection and matching steps as two interrelated problems, and some feedback between the matching and detection process could increase the accuracy of both individual steps.

One of the weaknesses of our SSVM and other machine-learning approaches for setting hyperparameters is that they require training data [51]. For future work, it may be interesting to see how such parameters may be set automatically (or at least reduce the amount of annotation the user needs to make), either by using contextual data from the images themselves, e.g. by automatically increasing regularization when the quality of the novel image worsens, as done in [52]. It should be noted, however, that once the method is trained, it can operate automatically and report uncertainty in the produced results. The user can then examine these results quickly (especially those highly uncertain ones) and approve or correct them; thus creating training data that can be used to re-train and improve the method.

Another limitation of our method is that it is not completely invariant to pose. Finding the globally optimal solution for fitting a deformable template (or atlas) or statistical shape model, with unknown pose (scale, rotation, and translation) remains an unsolved problem, despite recent progress [53, 54, 55]

Acknowledgements

This work was supported in part by a scholarship from Canadian Institutes of Health Research (CIHR) Skin Research Training Centre and by grants from the Natural Sciences and Engineering Research Council of Canada (NSERC) and Canadian Dermatology Foundation.

- [1] American cancer society (2009). cancer facts and figures 2009, american cancer society. 2009. Atlanta, USA.
- [2] R.P. Gallagher and D.I. McLean. The epidemiology of acquired melanocytic nevi- a brief review. *Dermatologic Clinics*, 13(3):595–603, 1995.
- [3] J. Gachon, P. Beaulieu, J. Sei, J. Gouvernet, J. Claudel, and M. Lemaitre M. Richard J. Grob. First prospective study of the recognition process of melanoma in dermatological practice. *Arch Dermatol*, 141(434–448):4, 2005.
- [4] H. Huang and P. Bergstresser. A new hybrid technique for dermatological image registration. *IEEE BIBE*, pages 1163–1167, 2007.
- [5] H. Mirzaalian, T. Lee, and G. Hamarneh. Uncertainty-based feature learning for skin lesion matching using a high order mrf optimization framework. volume 7511, pages 98–105, 2012.
- [6] H. Mirzaalian, G. Hamarneh, and T.K. Lee. Graph-based approach to skin mole matching incorporating template-normalized coordinates. *IEEE CVPR*, pages 2152–2159, 2009.
- [7] M.P. Segundo, C. Queirolo, O.R.P. Bellon, and L. Silva. Automatic 3D facial segmentation and landmark detection. pages 431–436, 2007.
- [8] P. Perakis, G. Passalis, T. Theoharis, and I. A. Kakadiaris. 3D facial landmark detection and face registration. 2010.
- [9] V. Potesil, T. Kadir, G. Platsch, and S. Brady. Personalization of pictorial structures for anatomical landmark localization. *IPMI*, pages 333–345, 2011.
- [10] C. Chena and W. Xie, J. Franke, P. Grutzner, L. Nolte, and G. Zheng. Automatic x-ray landmark detection and shape segmentation via data-driven joint estimation of image displacements. *MIA*, pages 1–13, 2014.
- [11] T. Sang, W. Freeman, and H. Tsao. A reliable skin mole localization scheme. *IEEE ICCV*, pages 1–8, 2007.
- [12] J.S. Pierrard and T. Vetter. Skin detail analysis for face recognition. *IEEE CVPR*, pages 1–8, 2007.
- [13] T. K. Lee, M.S. Atkins, M.A. King, S. Lau, and D.I. McLean. Counting moles automatically from back images. *IEEE TBE*, 52(11):1966–1969, nov. 2005.
- [14] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE PAMI*, 18(4):377–388, 1996.
- [15] C. Berg, L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. *IEEE CVPR*, pages 26–33, 2005.
- [16] M. Leordeanu, A. Zanfır, and C. Sminchisescu. Semi-supervised Learning and Optimization for Hypergraph Matching. *IEEE ICCV*, 2011.

- [17] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. *ECCV*, 5303:596–609, 2008.
- [18] Y. Zeng, C. Wang, W. Yang, X. Gu, D. Samaras, and N. Paragios. Dense non-rigid surface registration using high-order graph matching. *IEEE CVPR*, pages 382–389, 2010.
- [19] N. Ansari, M.-H. Chen, and E.S.H. Hou. Point pattern matching by a genetic algorithm. In *IEEE IECON*, volume 2, pages 1233–1238, 1990.
- [20] M. Tico and C. Rusu. Point pattern matching using a genetic algorithm and Voronoi tessellation. *EUSIPCO*, pages 1589–1592, 1998.
- [21] L. Zhang, W. Xu, and C. Chang. Genetic algorithm for affine point pattern matching. *Pattern Recognition Letters*, 24:9–19, 2010.
- [22] C. Xing, W. Wei, J. Wu, C. Zhang, and Y Zhou. Matching point pattern using a munkres genetic algorithm. *Journal of Convergence Information Technology*, 6(12):376–383, 2011.
- [23] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *IJCV*, 13:119–152, 1994.
- [24] A. Myronenko, X. Song, and M. Carreira-Perpinan. Non-rigid point set registration: Coherent point drift. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *NIPS*, pages 1009–1016. MIT Press, 2007.
- [25] S. Gold, A. Rangarajan, C. Lu, and E. Mjolsness. New algorithms for 2d and 3d point matching: Pose estimation and correspondence. *Pattern Recognition*, 31:957–964, 1998.
- [26] B. Jian and B. Vemuri. A robust algorithm for point set registration using mixture of gaussians. *IEEE ICCV*, 2:1246–1251, 2005.
- [27] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 24(4):509–522, 2002.
- [28] Y. Zheng and M.D. Doermann. Robust point matching for nonrigid shapes by preserving local neighborhood structures. *IEEE PAMI*, 28(4):643, 2006.
- [29] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.*, 89:114–141, 2003.
- [30] F. Escolano, E. Hancock, and M. Lozano. Graph matching through entropic manifold alignment. *IEEE CVPR*, pages 2417–2424, 2011.
- [31] B. Wyk and M. Wyk. A pocs-based graph matching algorithm. *IEEE PAM*, 26(11):526–530, 2004.
- [32] R. Zass and A. Shashua. Probabilistic graph and hypergraph matching. *IEEE CVPR*, pages 1–8, 2008.
- [33] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. *IEEE ICCV*, 2:1482–1489, 2005.
- [34] T. Cour, . Srinivasan, and J. Shi. Balanced graph matching. *NIPS*, pages 1–8, 2006.
- [35] Marius Leordeanu, Rahul Sukthankar, and Martial Hebert. Unsupervised learning for graph matching. *IJCV*, pages 1–18, 2011.
- [36] T. Caetano, J. McAuley, Cheng Q., and A. Smola. Learning graph matching. *IEEE PAMI*, 31(6):1048–1058, june 2009.

- [37] O. Duchenne, F. Bach, I. Kweon, and J. Ponce. A tensor-based algorithm for high-order graph matching. *IEEE TPAMI*, 1(99):1–13, 2011.
- [38] M. Chertok and Y. Keller. Efficient high order matching. *IEEE PAMI*, 32(12):2205–2215, 2010.
- [39] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. *Int. J. Comput. Vision*, 61(1):55–79, January 2005.
- [40] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural SVMs. *Machine Learning*, pages 27–59, 2009.
- [41] Douglas A. Perednia and Raymond G. White. Automatic registration of multiple skin lesions by use of point pattern matching. *CMIG*, 16(3):205–216, 1992.
- [42] Juha Roning and Marcel Riech. Registration of nevi in successive skin images for early detection of melanoma. *IEEE ICPR*, 1:352–357, 1998.
- [43] L. Breiman. Random forests. *Machine Learning*, pages 5–32, 2001.
- [44] T. Ojala, M. Pietikinen, and D. Harwood. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. *ICPR*, 1:582–585, 1994.
- [45] K. L. Vincken A. Frangi, W. J. Niessen and M. A. Viergever. Multiscale vessel enhancement filtering. *ICPR*, 1:130–137, 1998.
- [46] J. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.
- [47] T. Kohlberger, V. Singh, C. Alvino, C. Bahlmann, and L. Grady. Evaluating segmentation error without ground truth. *MICCAI*, pages 528–536, 2012.
- [48] R. Gallagher, J. Rivers, T.K. Lee, C.D. Bajdik, David I. McLean, and A.J. Coldman. Broad-Spectrum Sunscreen Use and the Development of New Nevi in White Children: A Randomized Controlled Trial. *JAMA*, 283(22):2955–2960, 2000.
- [49] N. Hu, R. Rustamov, and L. Guibas. Graph matching with anchor nodes: A learning approach. *IEEE CVPR*, pages 2906–2913, 2013.
- [50] H. Mirzaalian, T. K. Lee, and G. Hamarneh. Spatial normalization of human back images for dermatological studies. *IEEE JBHI*, pages 1–13, 2013.
- [51] C. Wang, N. Komodakis, and N. Paragios. Markov random field modeling, inference and learning in computer vision and image understanding: A survey. *Computer Vision and Image Understanding*, 117(11):1610–1627, 2013.
- [52] J. Rao, R. Abugharbieh, and G. Hamarneh. Adaptive regularization for image segmentation using local image curvature cues. *ECCV*, pages 651–665, 2010.
- [53] C. Wang, O. Teboul, F. Michel, S. Essafi, and N. Paragios. 3D knowledge-based segmentation using pose-invariant higher-order graphs. *MICCAI*, 6363:189–196, 2010.
- [54] D. Cremers, F. R. Schmidt, and F. Barthel. Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. *IEEE CVPR*, pages 1–6, 2008.
- [55] S. Andrews, C. McIntosh, and G. Hamarneh. Convex multi-region probabilistic segmentation with shape prior in the isometric logratio transformation space. *IEEE ICCV*, 8673:2096–2103, 2011.