A reliable mole localization scheme using cascaded filters

Taeg Sang Cho
Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139
taegsang@mit.edu

Abstract

This paper presents a novel framework to automatically detect and label moles on skin images in the presence of clutters, occlusions, and varying imaging conditions. The input image is processed with cascaded blocks to successively discard non-mole pixels. Our method first searches the entire input image for skin regions using a non-parametric skin detection scheme, and the detected skin regions are further processed using a difference of Gaussian (DoG) filter to find possible mole candidates of varying sizes. Mole candidates are classified as moles in the final stage using a trained support vector machine. To increase the mole classification accuracy, hair is removed if present on the skin image using steerable filters and a graphical model. Experimental results demonstrate a successful mole localization in varying imaging conditions.

1. Introduction

An emerging application of computer vision is to assist medical experts. A potential application ranges from a medical image database management system to an automatic disease diagnosis system [3, 5]. In this paper, a computer vision-based mole localization system is introduced, which can potentially be used in registering mole pattern changes automatically.

Mole pattern changes are important cues in detecting early signs of melanoma, a deadly skin cancer [5]. However, a principled system to count moles and record their patterns is lacking at the present. In fact, a daily routine of many dermatologists is to count moles and record changes for every patient, which can be both time-consuming and non-productive. The goal of the proposed system is to automate such routines to aid dermatologists.

While image processing techniques are extensively used in classifying moles as either malignant or benign [16, 21], quantifying mole patterns in a larger skin image has received less attention [9, 15]. In Lee *et al*. [9], moles from back torso images, taken under constrained imaging conditions, are localized using meanshift clustering and heuristic classification schemes. A related work by Pierrard and Vetter [12] is aimed at detecting skin irregularities, such as moles, for face recognition. Pierrard and Vetter modeled moles as black dots (Laplacian of Gaussian) on the skin surface, and classified the mole candidates using normalized cross correlation and the proposed saliency measure. While their approach is similar to ours in using scale-space filters to locate mole candidates, they capture only salient skin irregularities that are isolated from other irregularities. This aspect differs from our goal in that we are to locate all moles present in the image, not only the salient ones.

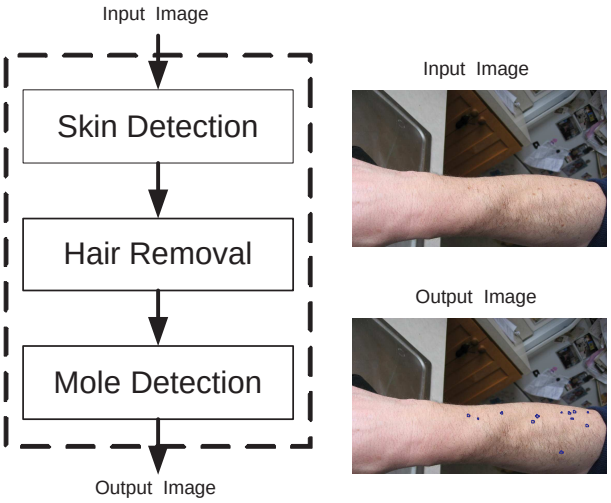


Figure 1. *Left:* A skin region is detected from the input image to reduce the computation and increase the reliability of the system. If any hair is present in the detected skin region, steerable filters are used to eliminate hair patterns. The hair removed image is further processed with the mole detection block to localize the moles. *Right:* Input - output image pair of the proposed system.

Our work extends upon Lee *et al*. to detect moles in skin images other than that of back torso under varying imaging conditions. A rich descriptor is used to classify moles using a support vector machine. To enhance the performance of the system, the input image is preprocessed to locate skin

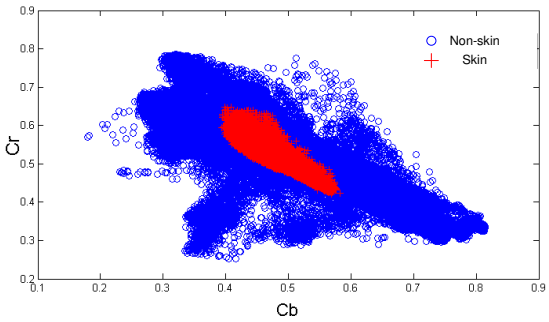


Figure 2. Skin color can be compactly localized in YCbCr color space [7]. This figure shows the skin and non-skin color distribution in CbCr space; the distributions shown serve as prior information about skin and non-skin pixels.

regions, and to remove hair if present in the detected skin region. Section 2 describes the developed system, and the purpose of each blocks.

2. Proposed System

Figure 1 shows a block diagram of the designed system. The system is comprised of three blocks: a skin detection block, a hair removal block, and a mole detection block. Since the input image can contain objects other than the skin surface, skin is detected to focus further image processing only on skin regions. If the detected skin region has hair patterns (user-specified), oriented filters are used to eliminate the hair patterns. The skin detection and hair removal steps reduce the computation and improve the mole detection performance. The hair removed images are used to locate possible mole candidates of different sizes using difference-of-Gaussian (DoG) filters [11]. Mole candidates are further classified as moles and non-moles using a trained support vector machine (SVM). Figure 1 shows an input-output pair of the skin image processed by the designed system.

3. Non-parametric skin region detection

Many successful skin region detection schemes have been proposed in literature [7, 8]; Vezhnevets *et al.* [18] provides an excellent review on methods using skin color as cues. However, many constraints are needed to use color as a reliable information source. For example, skin color can vary among races, and could vary even within the same person depending on the skin region of interest and lighting conditions. Also note the fact that the false positive rate could be high since many materials can have skin-like colors.

Fortunately, an exact segmentation of skin is not needed for our application: the subsequent image processing stages can be used to discard data from non-skin regions. Thus, our goal is to roughly estimate the skin area with small true negatives. Among many proposed skin detection methods, a non-parametric approach is taken since it's hard to find an adequate parametric representation of skin color data. YCbCr color space is used to represent skin colors since it's known that CbCr space localizes skin colors [7]. Figure 2 shows the distribution of skin color in our dataset, along with non-skin colors: skin color is confined in an elliptical distribution. The color distribution from the dataset is used as a prior skin color distribution. Note that it's important to have a representative skin dataset: we collected skin images from the web to capture different skin colors under varying lighting conditions. This tends to broaden the skin color distribution (Figure 2) compared with that in [7].

The skin pixel classification is carried out using a Neyman-Pearson hypothesis test: for a given pixel value x and a threshold λ, classify the pixel as skin if the ratio of probability being skin to probability being non-skin is greater than λ. In equation,

$$\frac{P_{X/H_1}(x/H_1)}{P_{X/H_0}(x/H_0)} \geq \lambda \qquad (1)$$

where H_1 is a hypothesis that the pixel is skin and H_0 is a hypothesis that the pixel is non-skin. Notice that $P_{X/H_1}(x/H_1)$ and $P_{X/H_0}(x/H_0)$ are the color distributions from the dataset. λ can be used to vary the operating point on the receiver operating characteristics (ROC) curve, and in this implementation, λ is fixed to be 1.

Artifacts of using only the pixel values, and no regional information, are salt-and-pepper type non-skin islands on inferred skin regions. These are removed by applying median filters on the detected skin regions, which results in a smooth skin mask. Another possible approach to remove non-skin islands is to adaptively change $lambda$ by looking at neighboring pixels. The threshold can be reduced if the neighboring pixels are classified as skin pixels and increased if the neighboring pixels are classified as non-skin pixels. However, the proposed adaptive scheme was not implemented for simplicity. Figure 3 shows some skin detection results: note that false alarms exist.

4. Hair removal using steerable filters and a graphical model

Hair can hinder a reliable mole detection, but we cannot ask users to shave hair before using the mole localization system. In this section, an image processing technique to remove hair patterns on skin images is described. There have been a number of successful attempts to remove hair patterns using a morphological erosion operator [10, 13], but these algorithms require that the hair patterns are in-focus, and the number of hair threads is small. Such constraints are not suited for our application since input images will be

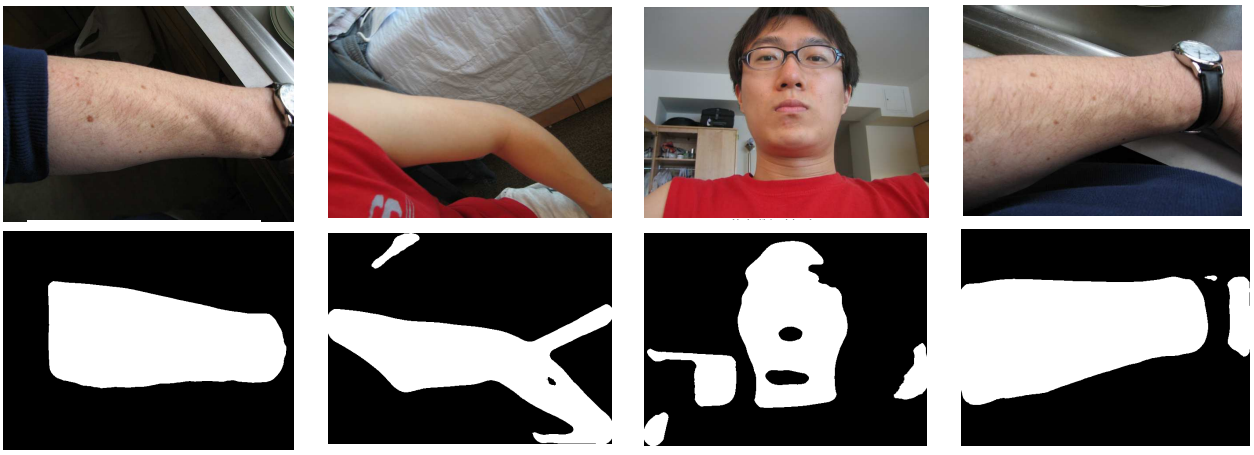


Figure 3. The skin region is detected using the skin color density distribution in Figure 2. Note that skin is not perfectly segmented, but this can be taken care of by the subsequent image processing stages.

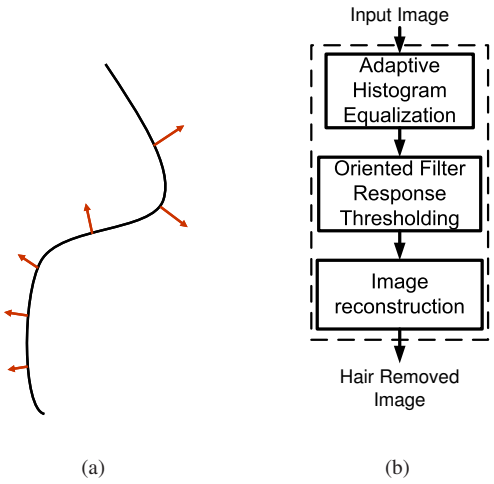


Figure 4. (a) Main idea behind the proposed hair removal scheme. Hair structures have a high derivative response at a direction normal to the hair orientation. Therefore, oriented filters can be used to extract out long hair-like structures. (b) The designed hair removal system. Hair is removed by thresholding the maximum magnitude of oriented derivative filters, and the skin image is reconstructed using a graphical model.

of larger skin regions with many hair threads. The proposed hair removal scheme solves this problem by using simple oriented filter operations.

The basic observation behind the hair removal scheme is that hair patterns have high frequency components normal to the orientation (Figure 4 (a).) Since hair patterns can be oriented arbitrarily, a large number of oriented filters would be required to search for the maximum normal derivative response at a given position. Once the maximum derivative response is calculated across the whole image, pixels with normal derivative magnitude greater than a certain threshold are classified as hair, and are discarded. The mathmatical expression of the proposed operation is given as the following:

$$\log(\max_{\phi}(F_{\phi}(x))) \geq \zeta(im) \tag{2}$$

where x is the intensity of the pixel, and F_{ϕ} is the magnitude response of derivative filter with orientation ϕ. Note that ζ is a function of the input image, and this will be explained more in depth.

In our implementation, a steerable filter [4] is used to reduce the number of oriented filters to 3. Filters introduced in [4] calculates the maximum derivative magnitude, as well as the direction (ϕ) that gives the largest magnitude. To enhance the distriminative power, the input image is adaptively histogram-equalized to accentuate the high frequency component of the input image.

The log of the derivative filter response at direction normal to hair orientation is rasterized into a histogram, denoted $H(x)$. This histogram apparently exhibits a normal distribution, so ζ in (2) is determined by the first and second order statistics of $H(x)$.

$$\zeta(im) = mean(H(x)) - \alpha \times std(H(x)) \tag{3}$$

Note that ζ is input image dependent. α is set to be 1 in this implementation, but can be adjusted automatically using machine learning techniques to account for the amount of hair present. The thresholded image will be missing many pixels: missing pixels are recovered by using a graphical model with a smoothness prior.

A graphical model is a compact representation of how multiple random variables interact with one another. If we assume that the underlying statistical process is Marko-

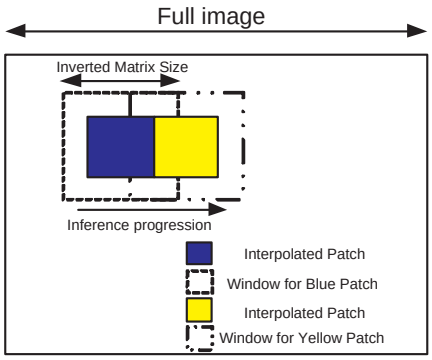


Figure 5. Only a windowed patch of J matrix is inverted for inference to reduce computational complexity. The blocky artifact of the proposed scheme is suppressed by using only the center portion of the interpolated patches.

vian and Gaussian, the graphical model is called a Gauss-Markov Random Field (GMRF) model. In GMRF, the maximum aposteriori (MAP) estimates and Bayes' least squares estimates equal $E[x|y]$. Assuming that x is a vector of underlying states, and y is a vector of observations through a noisy channel such that $y = Cx + v$, where $v \sim \mathcal{N}(0, R)$ and C is the selection matrix,

$$\hat{x}_{MAP} = \arg\max p(x|y) = J^{-1}h \tag{4}$$

where J is an information matrix and h is a mean inverse of the gaussian process. J and h encode how the state of a given node behaves as its neighbors and its measurement change, respectively. In this implementation, a thin-membrane model [2] is used as a smoothness prior for J matrix, which is given by (5).

$$\begin{aligned} p(x) &\propto \exp(-\alpha_2 \sum_{i \in V} (x_i - \frac{1}{|\mathcal{N}(x_i)|} \sum_{j \in \mathcal{N}(x_i)} x_j)^2) \\ &\equiv \exp(-x^T J_{prior} x) \end{aligned} \tag{5}$$

A thin membrane model minimizes the difference of neighboring node values.

In this implementation, every pixel is modeled as a node, and the whole image is represented as a mesh of nodes at the pixel level. Since the size of J matrix is the same as that of image, inverting J can be computationally expensive. While successful inference algorithms exist to solve the matrix inversion problem using belief propagation [19, 20], they often lead to inaccurate inference results for our graphical model.

In this work, an assumption is made that the potential of a given node will only depend on how the neighboring nodes behave: this is a Markovian property imposed the patch-level. This assumption can be justified in case of skin images since the skin surface does not contain a lot of high frequency components. Therefore, we can crop the image

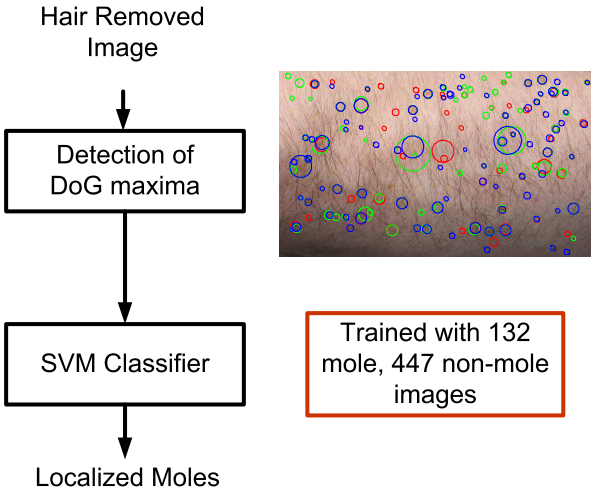


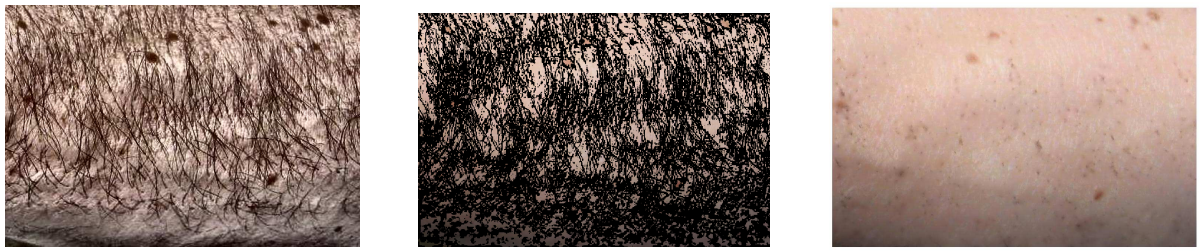
Figure 8. The hair removed image is run through a difference-of-Gaussian (DoG) filter in the scale-space. The maxima of the DoG filter output are considered possible mole candidates. The possible mole candidates are further classified into moles with a trained SVM classifier.

into smaller patches, as shown in Figure 5, and compute the inverse J matrix of the cropped window. Since the cropped patch is much smaller than the input image, it's computationally cheaper to invert the J matrix. The fact that we are doing the inference on blocks of image patches (i.e. ignoring pixels outside of window), the interpolated image will have blocky artifacts. Therefore, a window larger than the interpolated patch is used to calculate J^{-1}, and used only the center portion of the J^{-1} matrix, as shown in Figure 5.

Figure 6 shows how the image changes as it goes down the system pipeline, and Figure 7 shows the result of the proposed hair-removal scheme. The proposed scheme leaves moles in tact, while removing the hair pattern. Note that the hair removed image Figure 7 b) retains high frequency textures present in the input image. The size of moles may become a bit smaller as an artifact of using derivative filters. This can be somewhat relieved by using a smoothness prior in the inference stage. The performance of the proposed hair removal scheme is compared with that of Dull Razor ([10]). Since Dull Razor is to remove hair in high resolution, in-focus images with less hair threads, it performs poorly on our input image. In fact, Dull Razor misinterpretes moles as hair, and removes moles (Figure 7 c).

5. Mole localization using DoG filters and a support vector machine

The block diagram representation of the mole localization stage is shown in Figure 8. In this section, DoG scale-space filters and the designed support vector machine (SVM) classifier will be introduced.



(a) (b) (c)

Figure 6. (a) The output of the image after adaptive histogram equalization. Note that the high frequency components are accentuated. (b) The thresholding operation removes long trails of hair patterns, while leaving the mole patterns in tact. (c) A graphical model is used to interpolate the missing pixel values. To reduce the computational complexity of graphical model inference, a patch-wise Markovian property is assumed.

(a) (b) (c)

Figure 7. The input and output of the proposed system (a, b). The performance of the proposed scheme is compared with that of DullRazor [10].

Since the size of moles can vary, moles should be searched in a multi-scale fashion. SIFT feature detection scheme [11] solves the scale problem by using difference-of-Gaussian (DoG) filters in the scale-space. The same idea is used here: the DoG filter is applied to RGB color channels separately, and the set union of the output maxima in each channels are considered possible mole candidates. When combining the DoG maxima, any mole candidates occuring within a radius of another mole candidate is eliminated. Once mole candidates are localized, regions around mole candidates are cropped from the *hair-removed* image. The width of mole candidate patches is $2 \times \sqrt{2}$ times the radius of each DoG maximum.

Cropped mole candidates are classified as moles using a support vector machine (SVM) classifier. An SVM is a powerful tool to both generalize and classify objects: LIBSVM [1] is used to build the SVM classifier. The feature vector to train the SVM closely resembles that of Antonio Torralba *et al*. [17], and the procedure is delineated in Figure 9. A mole candidate patch is first resized to 32 x 32 using a bicubic interpolation scheme, and is converted into a LA*B* color representation. LA*B* components are then normalized as in (6) to increase the SVM classification performance.

$$\begin{aligned} L_{Norm} &= \frac{L}{100} \\ A_{Norm} &= \frac{A}{256} \\ B_{Norm} &= \frac{B}{256} \end{aligned} \tag{6}$$

The L component of the input patch is steerable filtered into a 2-scale steerable pyramid using MatPyrTools [14]: each scale consists of 6 different oriented filter outputs. There are 16 images at hand (12 filtered images, the low and high frequency residue of L, and A, B components of the input image), and they are each gridded into 4×4 squares. The values within the grid are averaged, and form an entry in the feature vector (i.e. each image generates 16 entries). The 16-entry vector from 16 images are rasterized into a single feature vector with 256 elements (Figure 9.) 132 mole patches and 447 non-mole patches are in the training set, and each mole patch is represented with a 256-entry feature vector described above. All mole patches come from *hair-removed* images, thus mole patches in the training set don't have any hair. Non-mole patches in the training set consist

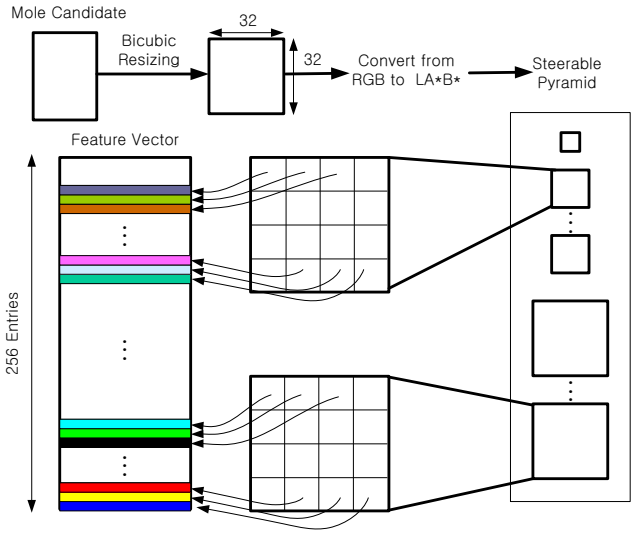


Figure 9. The mole candidate is resized into 32x32 patch image; the resized image is steerable filtered, and is represented as a 256-entry feature vector.

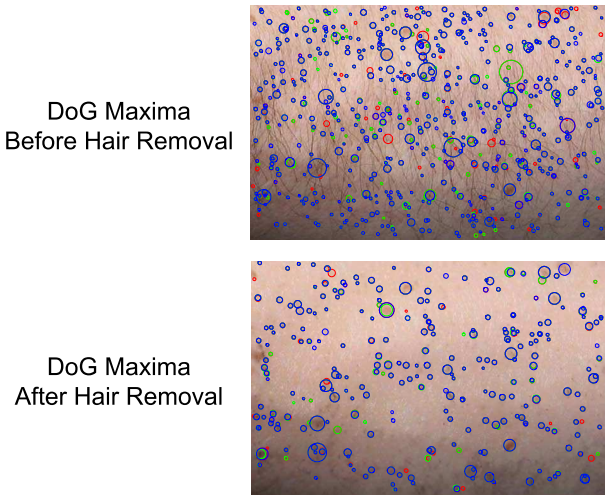


Figure 10. Hair removal scheme reduces the number of false mole candidates. The threshold of DoG response is adjusted in both images to get fewest DoG maxima while still retaining all moles.

of *non-mole* mole candidates from the DoG filters.

A standard dimensionality reduction scheme called Principal Component Analysis (PCA) is used to reduce the dimensionality of feature vectors. The number of principal components are chosen such that 98.5% of the data variance is conserved: 12 dimensions are sufficient with our dataset. The SVM training procedure is 10-fold cross-validated [6] to determine the optimal γ and C, which are two parameters of radial basis function-based SVM classifier.

6. Experimental results

To analyze how the hair removal scheme impacts the overall system performance, two sets of experiments are performed. In the first experiment, DoG filtering is applied to both the input image and the hair removed image. Experiments show that the hair removal scheme reduces the number of false mole candidates. Figure 10 shows the maxima of DoG filter outputs when hair is present and removed, respectively. The DoG filter in SIFT [11] discards maxima smaller than a certain threshold β. β is adjusted in both test cases to get least number of DoG maxima while retaining all moles. It's evident in this example that the hair removed image has fewer false mole candidates. Also, in other images, moles can be heavily occluded by hair such that the DoG cannot detect the mole pattern. In such cases, the hair removal scheme can help locate mole that would have been undetected otherwise.

In the second experiment, another type of SVM (called SVM 1 from here) is trained to quantify whether the hair removal scheme improves classification performance. SVM 1 is trained with mole patches that come from *skin surface*

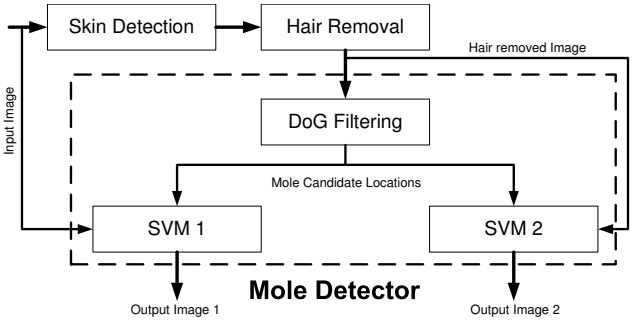


Figure 11. Another type of SVM is trained to verify the effectiveness of hair removal block in the system. While the mole candidate coordinates are the same for both SVMs, the SVM 1 crops the mole candidate patch from the hair removed image, while the SVM 2 crops the mole candidate patch from the original input image.

before hair removal. Thus, the mole patches in this training set have hair on them. This differs from our original SVM (called SVM 2 from here) in that SVM 2 is trained with hair-removed mole patches. The experiment architecture is shown in Figure 11. Inputs to both SVMs are the list of mole candidate locations, and the image to crop the mole patches from. SVM 2 crops the mole candidate patches from the hair removed image, whereas SVM 1 crops the mole candidate patches from the original image. Then, the output of each SVMs are compared to characterize the classification performance.

Figure 12 shows the result of the experiment. As can be seen, the designed system works well in the presence of varying arm orientations, poses and mole sizes. Although

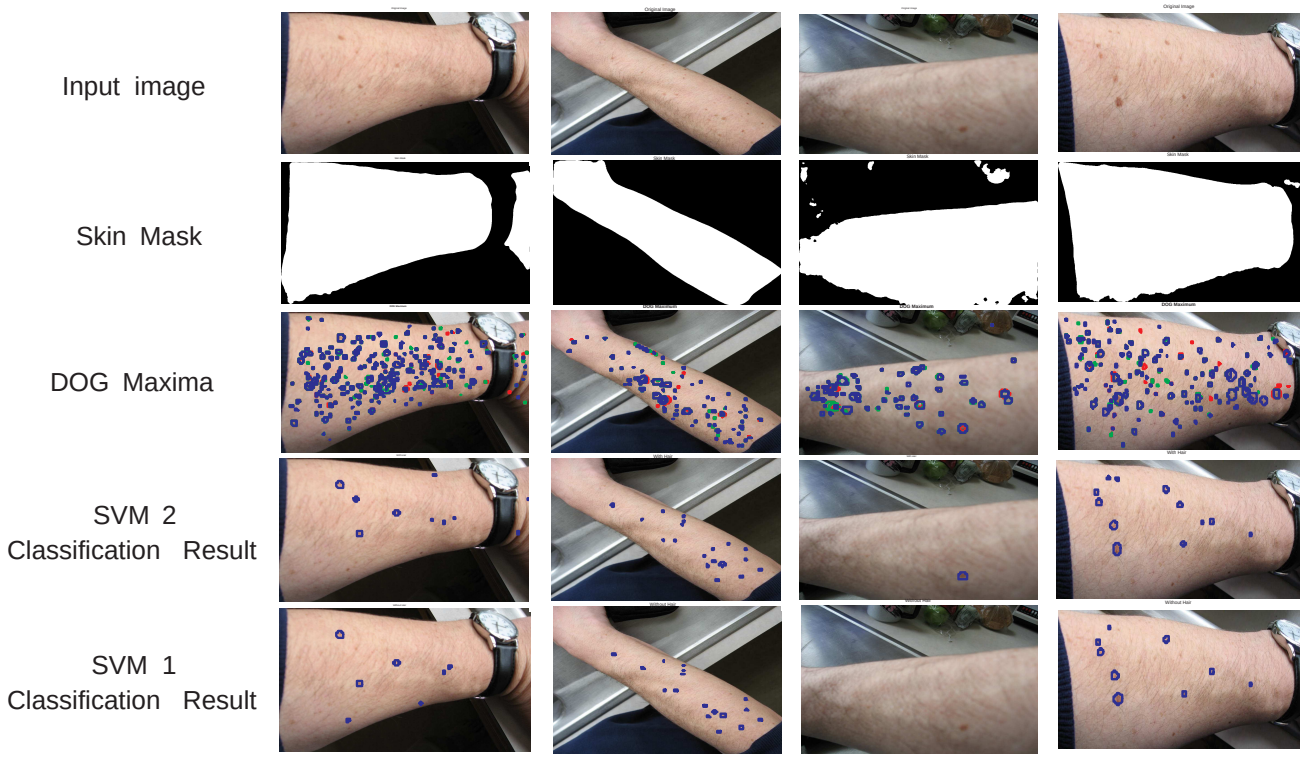


Figure 12. This figure shows the result of the experiment described in Figure 11. Notice that although the skin detection is not robust to objects with skin-like colors, the mole localization performance is quite robust. Interestingly, the mole localization performance in both SVMs are comparable, and the hair removal scheme doesn not drastically enhance the mole detection accuracy.

the skin detection method is not robust to objects with skin-like colors, subsequent stages can localize the moles reliably (i.e. at low false alarm rate) as long as skin regions are not discarded as non-skin region. From Figure 12, we can see that the hair removal scheme doesn't drastically enhance the mole detection accuracy. In some cases, SVM 1 does a better job in localizing moles, whereas in other cases, SVM 2 does a better job. To quantify the mole detection performance, two SVMs are tested with 10 different images. The detection result is summarized in Table 1.

In general, SVM 2 is more conservative than SVM 1. While SVM 1 has many false positives, it's detection rate is correspondingly higher than that of SVM 2. This can be due to two parameters of C-SVM (i.e. γ and C). From the experiment, we can safely infer that the hair removal doesn't drastically improve SVM's mole classification performance.

While both SVMs work well for most images, the system fails for image 7 and 8, shown in Figure 13. The high misclassification rate can be attributed to the fact that image 7 and 8 are somewhat blurry, and the moles on the skin surface are deformed due to affine transform. To increase the reliability, the input image should be in-focus, and moles should not be too deformed by affine transform.

Table 1. A table comparing the performance of SVM 1 and SVM 2 (TM: Total number of Moles in the image, TP: True Positive, FP: False Positive)

Image No.	TM	SVM 1		SVM 2	
		TP	FP	TP	FP
1	2	2	0	2	0
2	1	1	0	0	0
3	1	0	1	0	0
4	8	8	2	8	0
5	16	12	1	12	0
6	12	12	0	11	1
7	**14**	**5**	**2**	**2**	**1**
8	**10**	**1**	**0**	**0**	**0**
9	13	12	0	10	0
10	6	4	0	4	0

7. Conclusion

This paper presented a novel mole localization scheme that makes use of multiple cascaded filters. The explicit description of mole appearance using an introduced feature vector allows us to increase the mole detection accuracy at low false alarm rate. The failure modes can be circumvented by dealing only with focused images with small

Figure 13. These two example images show the failure mode of the designed system. When mole patches are blurred, and moles are deformed by an affine transform, SVM cannot recognize them as moles. Therefore, input images should be in-focus, with small affine deformation.

depth changes (i.e., affine deformation) along the skin surface. Future work will focus on characterizing the localizer's performance on a larger dataset, and develop methods to use the moles patterns to match skin images in an image retrieval system.

Acknowledgment

Authors would like to acknolwedge Dr. Stephan Dreiseitl for providing us with mole images, and Myung Jin Choi for interesting discussions and helpful remarks.

References

[1] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines. 2001.

[2] M. J. Choi. Multiscale gaussian graphical models and algorithms for large-scale inference. Master's thesis, Massachusetts Institute of Technology, 2007.

[3] S. Dreiseitl, L. Ohno-Machado, S. Vinterbo, H. Billhardt, and M. Binder. A comparison of machine learning methods for the diagnosis of pigmented skin lesions. *Journal of Biomedical Informatics*, 34:28 – 36, 2001.

[4] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38:587 – 607, 1991.

[5] H. Ganster, A. Pinz, R. Rohrer, E. Wildling, M. Binder, and H. Kittler. Automated melanoma recognition. *IEEE Transactions on Medical Imaging*, 20:233 – 239, 2001.

[6] C.-W. Hsu, C.-C. Chang, and C.-J. Lin. A practical guide to support vector classification.

[7] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:696, 2002.

[8] M. Jones and J. Rehg. Statistical color models with application to skin detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[9] T. Lee, S. Atkins, M. King, S. Lau, and D. McLean. Counting moles automatically from back images. *IEEE Transactions on Biomedical Engineering*, 52:1966 – 1969, 2005.

[10] T. Lee, V. Ng, R. Gallagher, A. Coldman, and D. McLean. Dullrazor: A software approach to hair removal from images. *Computers in Biology and Medicine*, 27:533–543, 1997.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60, 2:91 – 110, 2004.

[12] J.-S. Pierrard and T. Vetter. Skin detail analysis for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[13] P. Schmid-Saugeon, J. Guillod, and J.-P. Thiran. Towards a computer-aided diagnosis system for pigmented skin lesions. *Computerized Medical Imaging and Graphics*, 27:65 – 78, 2003.

[14] E. Simoncelli. Matlab pyramid tools. http://www.cns.nyu.edu/ lcv/software.html.

[15] J. Smolle, G. Parr, and A. Gerger. Skin mapping using a low-cost digital imaging. *Skin Research Technology*, 9:201, 2003.

[16] T. Tanaka, R. Yamada, M. Tanaka, K. Shimizu, M. Tanaka, and H. Oka. A study on the image diagnosis of melanoma. In *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, 2004.

[17] A. Torralba, A. Oliva, M. Castelhano, and J. M. Henderson. Contextual guidance of attention in natural scenes: The role of global features on object search. *Psychological Review*, 113(4):766 – 786, 2006.

[18] V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In *Graphicon*, 2003.

[19] M. Wainwright, T. Jaakkola, and A. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Transactions on Information Theory*, 45:1120–1146, 2001.

[20] M. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. Technical report, University of California, Berkeley, 2003.

[21] Z. Zhang, W. Stoecker, and R. Moss. Border detection on digitized skin tumor images. *IEEE Transactions on Medical Imaging*, 19:1128 – 1143, 2000.