BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **10**, No 3

Sofia • 2010

Image Enhancement Using Retinex Algorithms and Epitomic Representation

Violeta Bogdanova

Institute for Parallel Processing, 1113 Sofia
E-mail: vbogdan@bas.bg

Abstract: *The images taken with video cameras usually have uneven illumination and contain large amounts of noise. So their pre-processing is necessary to provide better visibility and better recognition when they are used in a Face Recognition System.*

This paper aims to present and to combine two methods for enhancing the images – the image epitomes and variants of Retinex algorithms. The image epitome is a relatively new method for image processing and the paper tests the ability of this method for noise reduction. The Retinex algorithms are used to balance the illumination and to eliminate the shadow.

The results demonstrate the image enhancement.

Keywords*: Retinex algorithms, epitomes, expectation-maximization.*

1. Introduction

The change in light conditions of the site leads to major changes in a two-dimensional image, which significantly degrade the operation of the algorithms for face recognition. Many studies have shown that changes in images of the same person due to uneven lighting are often greater than the difference in the images of different persons. To address this major problem in face recognition, the work was aimed at searching and finding effective methods for normalization of images under different lighting conditions. The paper investigates the Retinex methods.

For a noise reduction this paper presents a relatively new method – epitome images. The epitome of an image is a compact patch-based probabilistic model that contains both shape and texture information. This representation condenses an image into a smaller collection of patches that can be used to reconstruct the original image.

2. Image epitomes

The concept of image epitomes was proposed in 2003 by Jojic and Frey [1]. The epitome of an image is its miniature, condensed version containing the essence of the textural and shape properties of the image. The size of the epitome is considerably smaller than the size of the image or the object it represents, but the epitome still contains most of the constitutive elements needed to reconstruct the image. One epitome may be shared from a collection of images, e.g. when images are few consecutive frames from a video sequence, or when they are photographs of similar objects. The real image in a collection is determined by its epitome and a mapping from the epitome to the image pixels. In this way the original image can be reconstructed from the epitome, although not perfectly. The epitome model is useful for multiple vision applications, such as image classification, segmentation and denoising.

Each pixel in the epitome is modeled as a mixture of Gaussians. For each patch in the original image, the posterior probability of the mapping from each patch in the epitome to the image patch is calculated.

The epitome patch with the highest posterior probability of mapping is then used to generate that image patch in the reconstruction. It is desirable to use patches large enough to capture spatial properties of the image, but small enough to generalize across the image and across a collection of images.

To learn the epitome, the mapping probabilities and the epitome pixel means and variances are initialized, and the Expectation-Maximization algorithm (EM) is applied to update these values for either a set number of iterations or until convergence.

More formally, an epitome is a model of a given image that can be learned, starting from a set of sampled patches by specifying a generative model.

2.1. The epitome as a generative model of image patches

Assume that an image *X* consists of a set of patches $\{Z_k\}_{k=1}^{P}$. The patch shape can be arbitrary, but for simplicity, we assume that the patch shape is a square. The patches may be of various sizes and may overlap. One patch is a set of pixels, indexed by their position in the input image *X*, $Z_k = \{Z_{ik}\}$, where $Z_{ik} = X_i$. For each patch Z_k, the generative model uses hidden mappings T_k that the map coordinates $j \in E_k$ in the epitome to the coordinates $i \in S_k$ in the patch. An epitomic pixel contains two parameters, the mean μ_j and variance ϕ_j of the epitomic pixel. These variables are shown in Fig. 1.

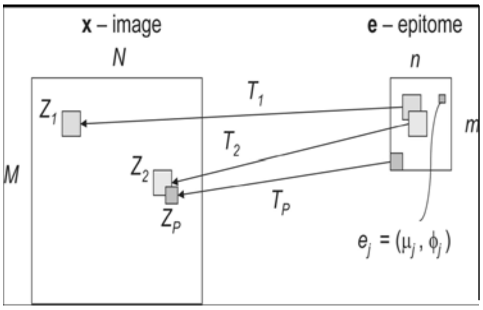

Fig. 1. A generative model of the epitome

Given the epitome $e = (\mu, \phi)$ and the mapping T_k, the patch Z_k is generated by copying the appropriate pixels from the epitome mean and adding Gaussian noise of a level given in the variance map. This can be formulated as Gaussian conditional probability.

For the patch Z_k, the conditional probability is defined as a product of the conditional probabilities of the pixels in this patch:

(1) $$p(Z_k \mid T_k, e) = \prod_{i \in S_k} N(Z_{i,k}\,;\, \mu_{T_k(i)},\ \phi_{T_k(i)})$$

where $N(Z_{i,k}\,;\ \mu_{T_k(i)},\ \phi_{T_k(i)})$ is a Gaussian distribution over $Z_{i,k}$ with mean $\mu_{T_k(i)}$ and variance $\phi_{T_k(i)}$.

For a collection of patches, the conditional probability is

(2) $$p(Z \mid T, e) = \prod_{k=1}^{P} p(Z_k \mid T_k,\ e)\,.$$

The graph representation of the epitome definition as a generative model of image patches is shown in Fig. 2.

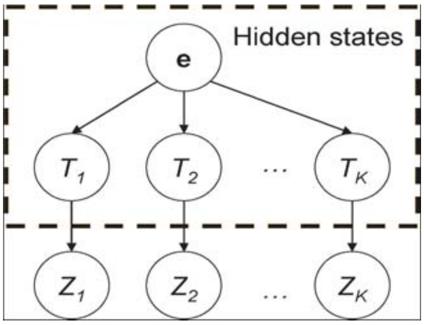

Fig. 2. Epitome as generative model of image patches

The patches are assumed to be generated independently, so the joint distribution is

(3) $$p(\left(\{\mathbf{Z}_k|T_k\}_{k=1}^{P}, e\right) =$$

$$= \prod_{k=1}^{P} p(\mathbf{Z}_k\,,\; T_k\,,\; e) = \prod_{k=1}^{P} p(\mathbf{Z}_k|T_k,\; e)p(T_k|e)p(e) =$$

$$= p(e)\prod_{k=1}^{P} p(T_k)p(\mathbf{Z}_k|T_k,\; e) \;=\; p(e)\prod_{k=1}^{P} p(T_k)\prod_{i\in Sk} N\left(z_{i,k}\,;\, \mu_{T_k(i)}\,,\, \phi_{T_k(i)}\right).$$

The equations are drawn using Bayes rule on conditional probabilities and assuming that the prior on all epitomes is flat and therefore does not appear in the parameter estimation.

The prior on the mappings $p(T_k)$ is used to take advantage of some mappings over others such as the choice of larger complete patches rather than several smaller components of the patch.

2.2. Learning the epitome model, using the EM algorithm

Learning the model requires to find the parameters $e=(\mu,\ \phi)$, that maximize the incomplete data likelihood $p(Z\,|\,e)$. This problem may be solved by using the complete data likelihood $p(Z\,,T\,|\,e)$ and the estimation of the parameters $\{T_k\}_{k=1}^{P}$ and *e* using the Expectation-Maximization (EM) algorithm.

The EM algorithm is an approximate way of estimating these parameters by iterating between an “expectation” E-step, finding the log-likelihood of the data from the complete log-likelihood, and a Maximization M-step, estimating the parameters from the log-likelihood.

The procedure of the EM algorithm is as follows.

1. The algorithm starts with an initial nontrivial guess at the configuration of the parameters.

2. The E-step involves taking the expectation of the complete log-likelihood $\log p(\{T\,|\,Z\}|\,e)$ over the hidden variables $\{T_k\}_{k=1}^{P}$ given the observed data $\{Z_k\}_{k=1}^{P}$ and parameters *e*.

The result from the E-step are the estimated mappings $\{\hat{T}_k\}_{k=1}^{P}$.

3. The M-step maximizes the resulting likelihood function of the data

$$L\left(e\,|\,\{\mathbf{Z}_k, T_k\}_{k=1}^{P}\right) = \log p\left(\{\mathbf{Z}_k\}_{k=1}^{P}\,|\,\{T_k\}_{k=1}^{P}, e\right).$$

The result from the E-step is the estimated epitome $\hat{e}$.

4. The parameters $\hat{e}=(\hat{\mu},\ \hat{\phi})$ found on the M-step at this iteration are used to begin the next E-step iteration and the process is repeated iteratively for a given number of iterations, or until convergence.

Following [5], the target function is given by

(4) $$Q(e,\ e^g) = E[\log p(Z,\ T \mid e)] = \sum_T \log p(Z,\ T \mid e) f(T \mid z,\ e^g).$$

Performing the EM iteratively, the posterior probabilities, based on the current parameters are computed at E-step:

(5) $$f(T_k \mid Z_k,\ e^g) = \frac{p(T_k) N(Z_k;\ \mu^g_{T_k},\ \phi^g_{T_k})}{\sum_{T_k} p(T_k) N(Z_k;\ \mu^g_{T_k},\ \phi^g_{T_k})}.$$

At M-step the estimated epitome means and variances are computed as follows:

(6) $$\mu_j = \frac{\sum_k \sum_{i \in S_k} \sum_{T_k, T_k(i)=j} f(T_k \mid Z_k,\ e^g) z_{i,k}}{\sum_k \sum_{i \in S_k} \sum_{T_k, T_k(i)=j} f(T_k \mid Z_k,\ e^g)},$$

$$\phi_j = \frac{\sum_k \sum_{i \in S_k} \sum_{T_k, T_k(i)=j} f(T_k \mid Z_k,\ e^g)(z_{i,k} - \mu^g_j)^2}{\sum_k \sum_{i \in S_k} \sum_{T_k, T_k(i)=j} f(T_k \mid Z_k,\ e^g)}.$$

At each iteration the likelihood with the newer set of parameters was proven to be greater than the likelihood of any previous iteration [6].

That is to say,

$$L\left(\theta^{(0)} \mid z\right) > L\left(\theta^{(1)} \mid z\right) > \ldots > L\left(\theta^{(t-1)} \mid z\right) = L\left(\theta^{(t)} \mid z\right)$$

where $\theta = e$, $z = \{\mathbf{Z}_k, T_k\}_{k=1}^{P}$ in the case of epitome estimation and the superscript is the iteration number.

The advanced idea for image epitome is developed for video epitomes [2].

3. Retinex algorithm

Retinex is an image enhancement algorithm that is used to improve the contrast, brightness and sharpness of an image primarily through dynamic range compression. The algorithm also simultaneously provides color constant output and thus it removes the effects caused by different illuminants on a scene. It synthesizes contrast enhancement and color constancy by performing a non-linear spatial/spectral transform. The original algorithm is based on a model of human vision's lightness and color constancy developed by Edward Land. Jobson et al. extended the last version of Land's model [7-10] and have since added several improvements to the original version of Retinex, including the use of multiple scales, color restoration, and some other variants.

Retinex belongs to the class of center surround functions, where each output value of the function is determined by the corresponding input value (center) and its neighborhood (surround). For Retinex the center is defined as each pixel value and the surround is a Gaussian function. There are some variants of this algorithm.

3.1. Single scale retinex algorithm

The Single Scale Retinex (SSR) [7] is defined for a point (x, y) in an image as:

(7) $$R_i(x, y) = \log I_i(x, y) - \log[F(x, y) * I_i(x, y)], \qquad i = 1, \ldots, S,$$

where the sub-index i represents the i-th spectral band, S is the number of spectral bands (S =1 for grayscale images, and S =3 for typical color images);

$R_i(x, y)$ is the Retinex output and $I_i(x, y)$ is the input image distribution in the i-th spectral band. The symbol “*” denotes the convolution operation;

$F(x, y)$ is the normalized surround function; various surround functions could be used, and the Gaussian surround function is one of them,

(8) $$F(x, y) = Ke^{-(x^2+y^2)/c^2}$$

where c is the Gaussian surround constant, that is referred to as the scale of the SSR, and K is selected such that

$$\iint F(x, y)\, dxdy = 1.$$

A small value of c provides a good dynamic range compression, and a large scale provides better colour rendition.

The image distribution is the product of scenes reflectance and illumination:

(9) $$I_i(x, y) = S_i(x, y)\, r_i(x, y)$$

where $S_i(x, y)$ is the spatial distribution of illumination, and $r_i(x, y)$ is the distribution of scene reflectances.

$$R_i(x, y) = \log[S_i(x, y)\, r_i(x, y) / (\overline{S_i(x, y)\, r_i(x, y)})].$$

As the illumination generally has slow spatial variation $S_i(x, y) \approx \overline{S_i(x, y)}$, then:

(10) $$R_i(x, y) = \log[r_i(x, y) / \overline{r_i(x, y)}].$$

Equation (10) means that color constancy (independence from source illumination spectral and spatial variation) is achieved.

3.2. Multi-scale retinex algorithm

The Multi-Scale Retinex algorithm MSR is an extended SSR with the multiple kernel windows of the different sizes. The output of MSR is a weighted sum of several different SSR outputs [8].

The Multi-scale retinex algorithm is given by

$$R_i\,(x,y)=\sum_{n=1}^{N} w_n\,R_{ni}, \qquad i=1,\ldots,S, \tag{11}$$

where

$$R_{ni}(x,y)=\log I_i(x,y)-\log\left[F_n(x,y)*I_i(x,y)\right],$$

$$R_i(x,y)=\sum_{n=1}^{N} w_n\,(\log I_i(x,y)-\log(F_n(x,y)*I_i(x,y)), \qquad i=1,\ldots,S,$$

N and S are the number of scales, and the number of spectral bands, respectively.

Here $R_{ni}(x,y)$ denotes a retinex output associated with the n-th scale for an image, $I_i(x,y)$ and $F_n(x,y)$ denote a surround function.

Note that a gain w_n is set to satisfy the condition $\sum_{n=1}^{N} w_n=1$. The surround function is given by: $F_n(x,y)=K_n e^{-(x^2+y^2)/c_n^2}$, where c_n are the scales that control the extent of the surround (smaller values of c_n lead to narrower surrounds), and $K_n=1/(\sum_x \sum_y F(x,y))$ is the normalization factor.

The obvious question about MSR is the number of scales needed, the scale values, and the weight values. The experiments have shown that three scales are enough for most of the images, and the weights can be equal. Generally fixed scales of 15, 80 and 250 can be used. But these are more experimental than theoretical, because we do not know the scale of an image to the real scenes. The weights can be adjusted to weight more on dynamic range compression or colour rendition.

4. Results

The paper examines at first the epitome properties in denoising. Denoising is still one of the most fundamental tasks in image processing. Despite the fact that it is widely studied, there are many unsolved problems yet. The purpose of denoising is to estimate the original image (or a “better” representative of it) from the noisy data.

To illustrate the potential power of the epitomes, the algorithm was tested by images with a high level of multiplicative noise and Gaussian noise. The epitome model of an image was used to improve the images. It is important to know that there are additional ways to tune the algorithm as properly choosing the size of patches, the number of iterations and the initial point for the EM algorithm.

Figs. 3(a), 3(b), 3(c), 3(d) and Figs. 4(a), 4(b), 4(c) demonstrate the Epitome algorithm for image denoising.

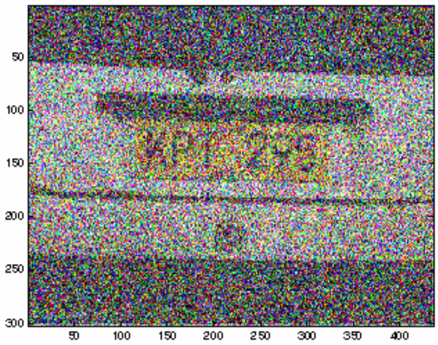

Fig. 3(a). Noisy image (30×400)

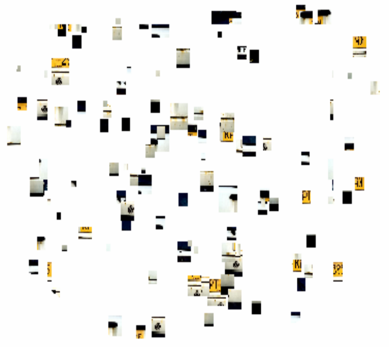

Fig. 3(b). A set of patches

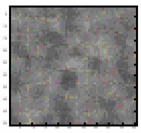

Fig. 3(c). Epitome (50×50)

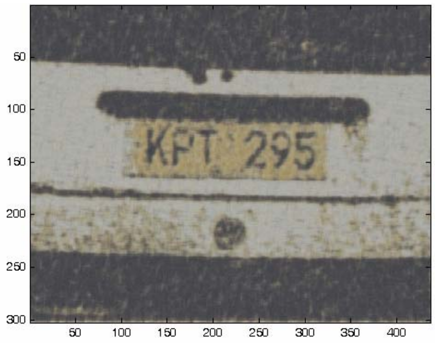

Fig. 3(d). Denoised image (300×400)

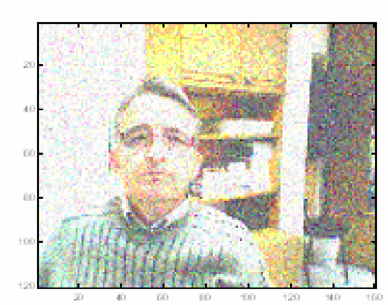

Fig. 4(a). Noisy image (120×160)

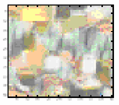

Fig. 4(b). Epitome (50×50)

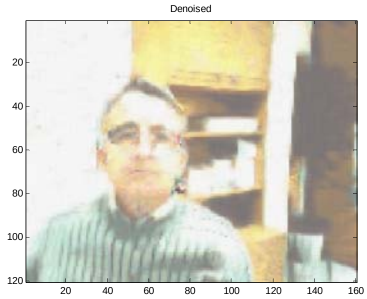

Fig. 4(c). Denoised image (120×160)

The second step is to examine the Retinex algorithms.

Figs. 5(a), 5(b), 5(c) and 6(a), 6(b), 6(c) demonstrate the Single Scale Retinex algorithm and the Multi Scale Retinex algorithm for two images (the second image is from YaleB data-base). It is seen (a) that the illumination is very uneven. The improved images (c) then are used as input images from a Face Recognition System

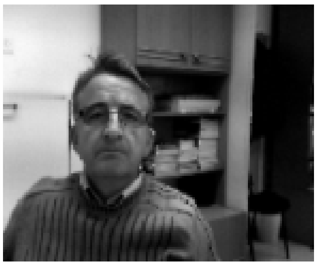

Fig. 5(a). Input image

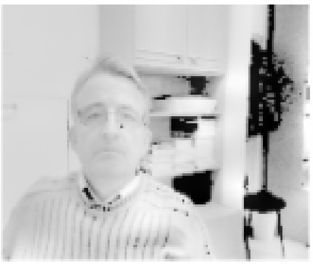

Fig. 5(b). Image after SSR (c=10)

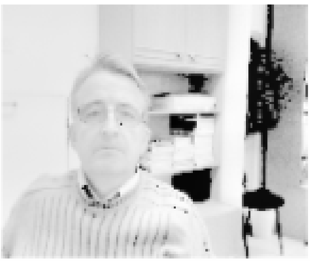

Fig. 5(c). Image after MSR (10, 80, 250)

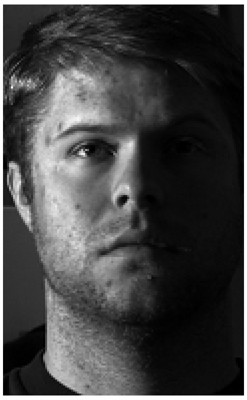

Fig. 6(a). Input image

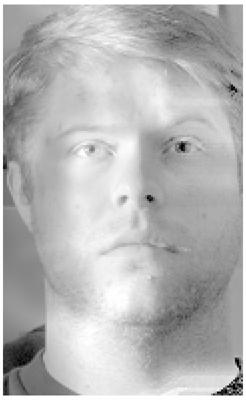

Fig. 6(b). Image after SSR (c=10)

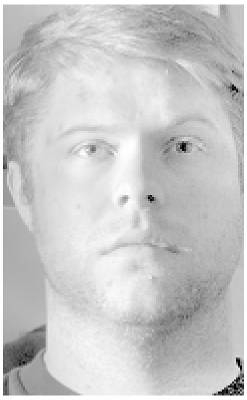

Fig. 6(c). Image after MSR (10, 80, 250)

5. Conclusion

The study explores the combination of two methods for pre-processing of images.

The paper describes the epitomes as recent models of appearance and shape, composed of image patches. The epitome of an image is found, using a variant of EM algorithm. This paper shows most of the steps in deriving the update equations for the epitome as a generative model of image patches. The paper investigates the use of the epitomic representation and reconstruction for filtering very noisy images.

The Retinex algorithms are tested on a real image, obtained from a camera in uneven lighting. The results show very good reducing of the impact of uneven illumination and the presence of shadows and reducing the noise.

The results are intended and tested as input images in a real system for face recognition and the results are encouraging at this stage of system development, since better face recognition of pre-processed images is achieved.

Acknowledgements: This work is partially supported by the the Bulgarian National Science Fund under grant No D002-240/18.12.2008 "Study and Modelling of the Age-related Changes in Visual Information Processing", and by IICT Project No 2 "Design of Software for Collecting and Hierarchical Processing of Information from Hundreds of Sensors".

References

1. Jojic, N., B. Frey. Epitomic Analysis of Appearance and Shape. – In: Proc. of ICCV, 2003.
2. Cheung, V., B. J. Frey, N. Jojic. Video Epitomes. – In: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, 2005.
3. Frey, B., N. Jojic. Learning the Epitome of an Image. University of Toronto, TR PSI-2002-14, November 10, 2002.
4. Dempster, A. P., N. M. Laird, D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. – Journal of the Royal Statistical Society, 1977.
5. McLachlan, G., T. Krishnan. The EM Algorithm and Extensions. New York, John Wiley & Sons, 1996.
6. Jobson, D. J., Z. Rahman, G. A. Woodell. Properties and Performance of a Center/ Surround Retinex. – IEEE Transactions on Image Processing, Vol. **6**, 1997, No 3, 451-462.
7. Hines, G. D., Z. Rahman, D. J. Jobson, G. A. Woodell. Single-Scale Retinex Using Digital Signal Processors. – In: Global Signal Processing Expo (GSPx), 2004.
8. Jobson, D. J., Z. Rahman, G. A. Woodell. A Multiscale Retinex for Bridging the Gap Between Color Images and the Human Observations of Scenes. – IEEE Transactions on Image Processing, Vol. **6**, 1997, No 7, 965-976.
9. Rahman, Z., D. J. Jobson, G. A. Woodell, G. D. Hines. Image Enhancement, Image Quality and Noise. Photonic Devices and Algorithms for Computing VII. – In: Proc. SPIE 5907, 2005.
10. Rahman, Z., D. J. Jobson, G. A. Woodell. Retinex Processing for Automatic Image Enhancement. – Journal of Electronic Imaging, January 2004.
11. Chen, W., M. J. Er, S. Wu. Illumination Compensation and Normalization for Robust Face Recognition Using Discrete Cosine Transform in Logarithmic Domain. – IEEE Transactions on Systems, Man and Cybernetics. Part B. Vol. **36**, 2006, No 2, 458-466.