Transferring Colours to Grayscale Images by Locally Linear Embedding

Jun Li, Pengwei Hao
Department of Computer Science, Queen Mary, University of London
Mile End, London, E1 4NS
{junjy, phao}@dcs.qmul.ac.uk
Chao Zhang
Center for Information Science, Peking University
Beijing, 100081
chzhang@cis.pku.edu.cn

Abstract

In this paper, we propose a learning-based method for adding colours to grayscale images. In contrast to many previous computer-aided colourizing methods, which require intensive and accurate human intervention, our method needs only the user to provide a colourful image of the similar content as the grayscale image. We accept the "image manifold" assumption and apply manifold learning methods to model the relations between the chromatic channels and the gray levels in the training images. Then we synthesize the objective chromatic channels using the learned relations. Experiments show that our method gives superior results to those of the previous work.

1 Introduction

Colours make images more vivid. They can now be recorded with a point-and-shoot camera easily. However, people often want to add colours to old monochrome photos, and pictures are sometimes shot with severely wrong white balance settings, in such a case, a possible remedy is to keep only the captured intensities and transfer colours from another source to it. The technique of adding colours is particularly useful when the image is taken with special sensors, such as X-ray, MRI, near infrared and so on. This is called "pseudo-colouring". The difficulty of assigning colours to a monochrome image rises from the lack of deterministic relations between the luminance and the hue/saturation channel of an image – in an image, pixels of the same intensity may have different colours and vice versa.

A human may intuitively guess the colours given a monochrome picture, because we know what are in the picture and we have prior knowledge on how their colours should be. However, assigning colours to each pieces in an image is very tedious [10]. In contrast to that for a human, the task of recognizing a scene or a subject is difficult to a machine, while the task of transferring colour is a tractable low-level vision task.

In this paper, we adopt the idea in the semi-automatic scheme by Welsh et al. [13] and propose a learning-based method to colourize grayscale images. In our framework, the

BMVC 2008 doi:10.5244/C.22.83

user assists the algorithm by selecting an image that contains similar contents to the input grayscale image. Then the algorithm cut both images into overlapping patches, which are assumed to be distributed on a manifold ([5; 1]). For each input patch, we find its neighborhood in the training patches and infer its chromatic information by the manifold learning method of locally linear embedding ([8]).

After a brief background review in Section 2, we analyze the problem and present our framework in Section 3. In Section 4, we show experimental results, and report our experiment configurations as well. Finally, we conclude the paper in Section 5.

2 Related Works

Computer-assisted systems for colourizing pictures have been studied in the cartoon industry. However, they often require tedious human work. Qu et al. present an algorithm that do the task with very human intervention [7]. However, this technique still requires the user to approximately specify the colours in each part of the image. This lowers the efficiency and prevents the user from processing multiple images in one time. Our work is inspired by the idea proposed by Welsh et al. [13], in which the system asks the user for a training image with the similar content to the grayscale image to be colorized, and transfers the colours from the training image to the grayscale image. Welsh's algorithm assigns an input pixel with the colour of a training pixel, which is found by matching a small local patch in the intensity channel of the training image. We develop this idea by adopting the assumption that the patches are lying on two image manifolds [1; 9] and employing manifold learning algorithm [8] to estimate the colours. Our work can be seen as extending the applications of manifold learning [6] to low-level vision tasks [2; 5] to a new problem of inferring chromatic information.

3 Colorizing Images with LLE

3.1 Problem Analysis

The problem of transferring colour from chromatic images to a monochrome image can be stated formally as follows: Given a monochrome image $\mathbf{X}_t$, and a training image, $\mathbf{X}_s$ and $\mathbf{Y}_s$, where $\mathbf{X}_s$ denotes the intensity channel, and $\mathbf{Y}_s$ denotes the chromatic channel, we want to discover the relations between $\mathbf{X}_s$ and $\mathbf{Y}_s$ so that we can estimate the chromatic channels $\mathbf{Y}_t$ for $\mathbf{X}_t$. Note that in our denotion, "x" is for the monochrome side and "y" is for the chromatic side; and in the subscript, "t" is for the input and target image and "s" is for the training image.

Represented by the pixel values, images are lying in a very high-dimensional space, which hampers discovering the meaningful properties of their distribution. While for many low-level vision tasks including ours, each pixel in the image can be safely considered related to only a few neighbouring pixels. Therefore, we represent our images using overlapped patches. We use the following denotations: $\mathbf{X}_t := \{\mathbf{x}_t^p\}_{p=1}^{N_t}$, $\mathbf{Y}_t := \{\mathbf{y}_t^p\}_{p=1}^{N_t}$, $\mathbf{X}_s := \{\mathbf{x}_t^q\}_{q=1}^{N_s}$ and $\mathbf{Y}_s := \{\mathbf{y}_s^q\}_{q=1}^{N_s}$. N_t and N_s are the number of patches in the input image and the training images respectively.

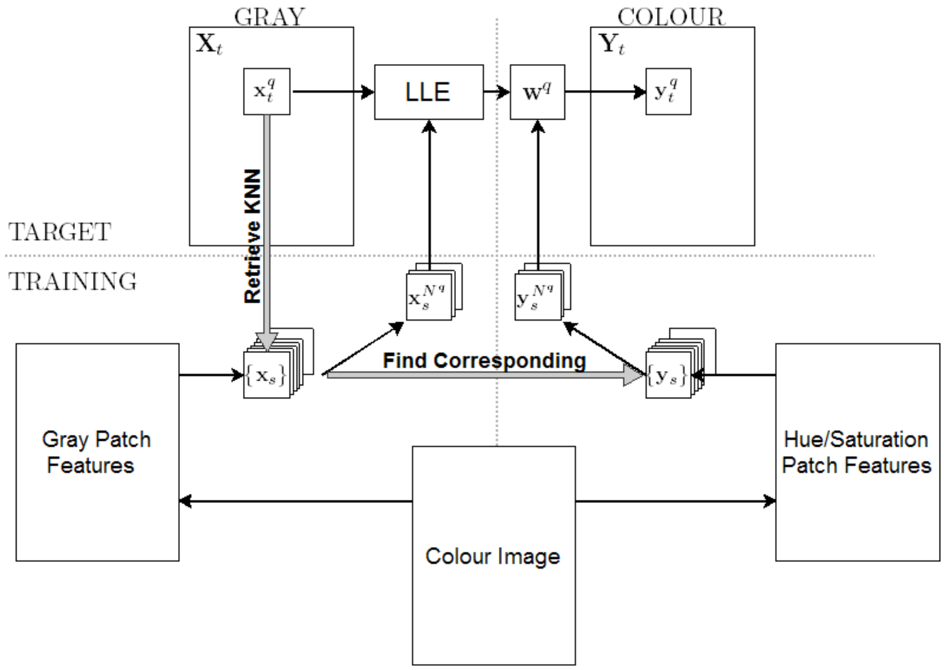


Figure 1: Flow Chart of Colourization with LLE

3.2 Manifold Learning

We adopt the assumption pointed out in the previous research [4; 12; 11; 2; 5] that the pixel values in a patch are controlled by only a few factors, thus they are distributed in a (generally non-linear) low-dimensional manifold. We can also assume that based on the same information source, the manifolds of the grayscale patches and the chromatic patches share similar geometric structures, although they are embedded in distinct spaces. To make use of the shared geometric structures and predict the unknown chromatic patch for an input grayscale patch, we adopt the *locally linear embedding* (LLE) algorithm [8].

LLE is designed to find a low-dimensional representation for a set of manifold samples embedded in a high-dimensional space. For the set $\{\mathbf{u}^p\}$ it computes their low-dimensional coordinates $\{\mathbf{v}^p\}$ by optimization such that: if each $\mathbf{u}$ is approximated by linear combination of its K nearest neighbours, using the same combining coefficients and neighbours for each $\mathbf{v}$, the $\{\mathbf{v}^p\}$ minimizes the residual.

3.3 Estimating the Colours

The LLE can not only model the relation between high dimensional embedded manifold points and their low dimensional parameterizations but also model the relation between two structurally similar manifolds. In our framework, given an input grayscale patch $\mathbf{x}_t^q \in \mathbf{X}_t$, we estimate its chromatic patch $\mathbf{y}_t^q$ as follows:

1. Find K nearest neighbours of $\mathbf{x}_t^q$ in $\mathbf{X}_s$, denote their indices as $\mathbf{N}^q$.
2. Compute a coefficient vector $\mathbf{w}^q$. Such that combining the neighbours $\{\mathbf{x}_s^r\}_{r\in\mathbf{N}^q}$ with $\mathbf{w}^q$ minimizes the residual for $\mathbf{x}_t^q$.
3. Synthesize $\mathbf{y}_t^q$ by combining the corresponding neighbours $\{\mathbf{y}_s^r\}_{r\in\mathbf{N}^q}$ with $\mathbf{w}^q$.

In step 2, $\mathbf{w}^q$ is found:

$$\mathbf{w}^q = \operatorname*{argmin}_{\mathbf{w}} \|\mathbf{x}_t^q - \sum_r \mathbf{w}(r)\mathbf{x}_s^r\|_2^2 \tag{1}$$

subject to

$$\sum_r \mathbf{w}^q(r) = 1 \tag{2}$$

$$\mathbf{w}^q(r) = 0, \text{ for } r \notin \mathbf{N}^q \tag{3}$$

The optimal $\mathbf{w}^q$ in Eq(1) can be readily found by by solving a linear system [8]. In Figure 1, we draw a flowchart for our framework of colourization.

4 Experiment

4.1 Representing Patches in Feature Vectors

The grayscale feature vectors in $\mathbf{X}_s$ and $\mathbf{X}_t$ for each patch are consisting of three components: the average intensity value, the first and the second order derivatives [2; 3].

Consider the intensity of an image as a function $\mathscr{I} : \mathbb{Z}^2 \to \mathbb{R}$. Let ∇_x and ∇_y represent the horizontal and vertical differentiating operators respectively:

$$\nabla_x \mathscr{I}(x,y) = \mathscr{I}(x+1,y) - \mathscr{I}(x-1,y) \tag{4}$$

$$\nabla_y \mathscr{I}(x,y) = \mathscr{I}(x,y+1) - \mathscr{I}(x,y-1) \tag{5}$$

Then for a grayscale patch $\mathbf{P}$, we construct the feature vector

$$\left[\begin{array}{ccccc} \lambda\overline{\mathscr{I}|_{\mathbf{P}}} & \nabla_x \mathscr{I}|_{\mathbf{P}} & \nabla_y \mathscr{I}|_{\mathbf{P}} & \nabla_x^2 \mathscr{I}|_{\mathbf{P}} & \nabla_y^2 \mathscr{I}|_{\mathbf{P}} \end{array}\right]^T \tag{6}$$

where

$$\overline{\mathscr{I}|_{\mathbf{P}}} = \frac{\sum_{(x,y)\in\mathbf{P}} \mathscr{I}(x,y)}{|\mathbf{P}|}$$

and λ is weight of the intensity. It is chosen according to the patch size. It is needed, because however many pixels a patch has there is only one entry in the feature vector for the intensity. The objective feature vectors in $\mathbf{Y}_s$ are simply the pixel values in the chromatic patches.

In Figure 2, we show an input patch and its 5 nearest neighbors. In the figure, (a) and (b) are the input and training images. We show an example in which the first 3 nearest patches are found in (b) for a query patch (the square on the top of a tree on left) in (a).

In the patch table, the leftmost column shows the query (input) patch and its features. Each of the following column shows a neighbour and the corresponding features. Row

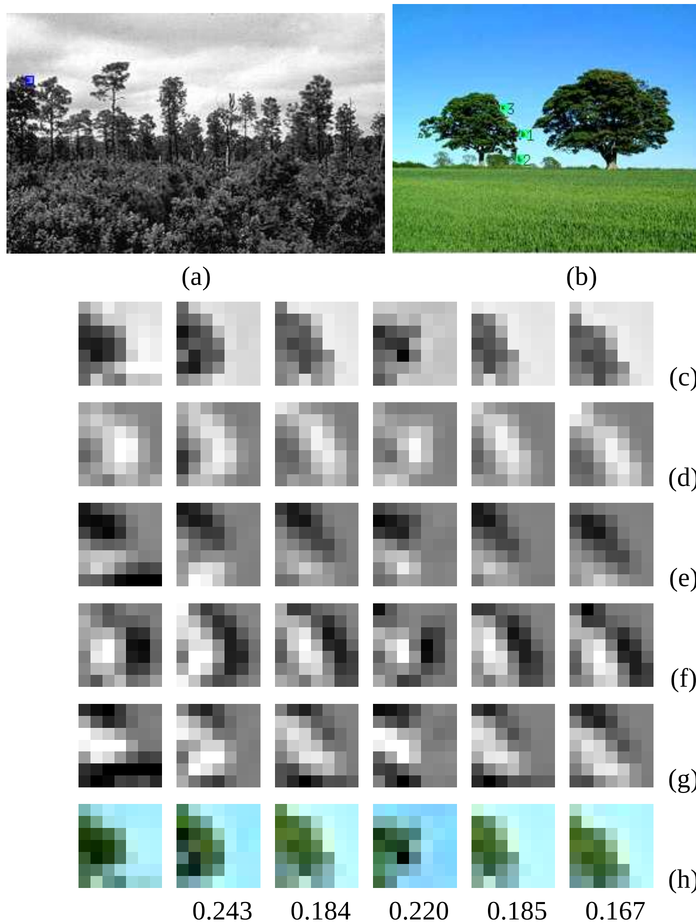


Figure 2: Nearest neighbors and reconstruction process
(a) input; (b) training; (c) – (g) features; (h) synthesis

(c) shows the grayscale patches. (d) and (e) show the first order horizontal and vertical gradients respectively. (f) and (g) show the second order gradients. Row (h) shows the colourful training patches and the synthesized colourful patch for the input patch. The reconstructing coefficients are listed below each training patch.

4.2 Experiment Results

In Figure 3, we test our algorithm on the images from a figure in [13]. The training and input images are those we have shown in Figure 2. In Figure 3, both (a) and (b) are generated by Welsh's algorithm. The image in (a) is generated by the algorithm with *global matching* ([13]). While to generate the image in (b), the algorithm was provided where are the sky and plant areas manually. In (c), we show our algorithms result. We do not need manual region matching. Note that in image (b), there are still pixels in the

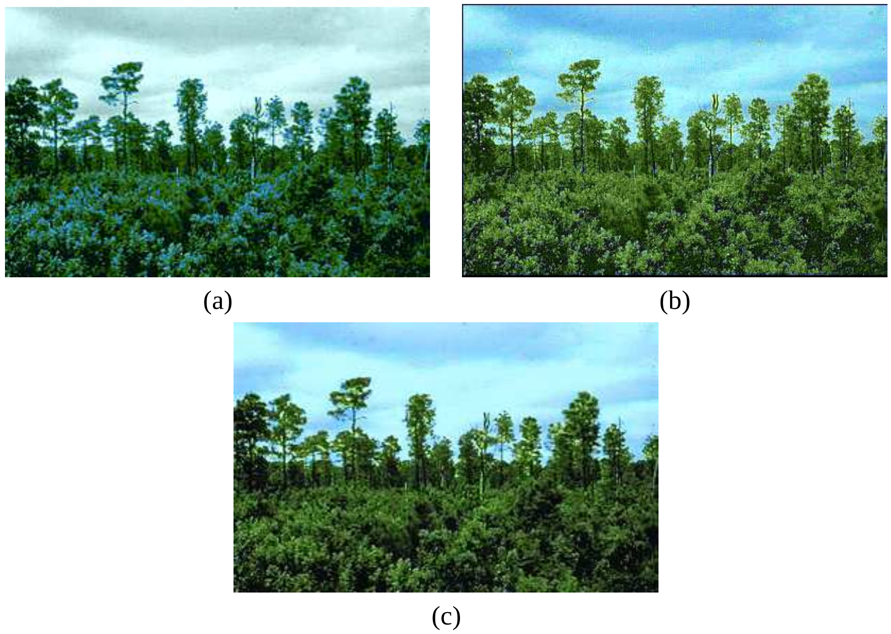


Figure 3: Colourization of Landscape Image I
(a) synthesized as in [13], global matching (b) as (a), with manual region assignment; (c) ours

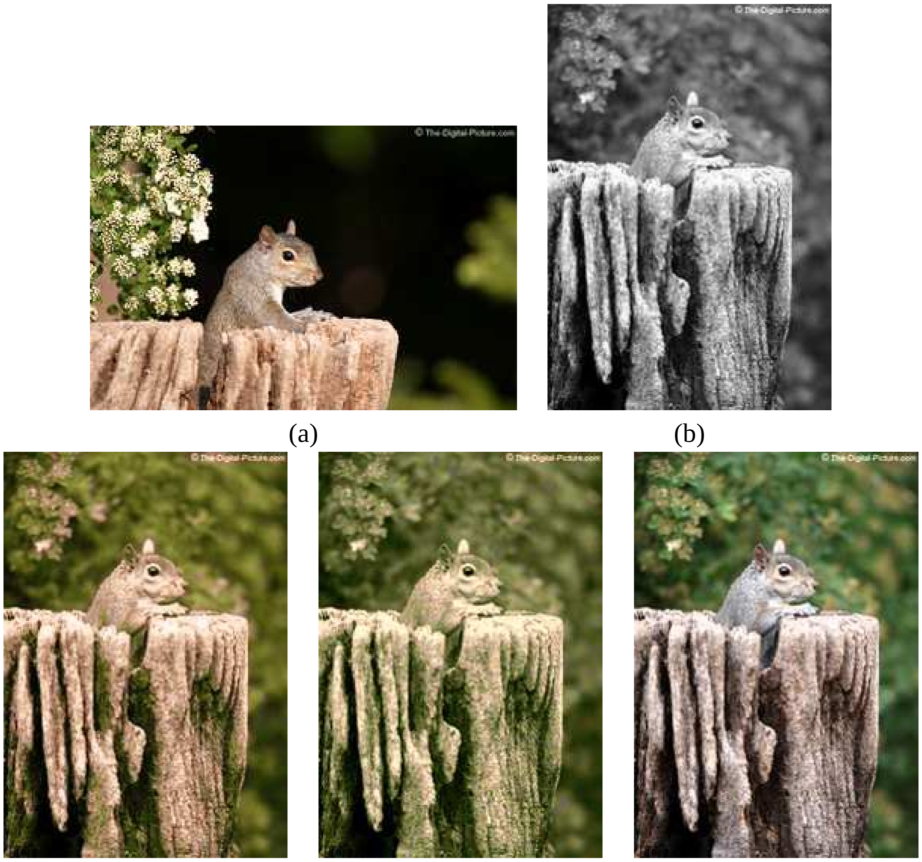
jungle area with wrongly assigned cyan/blue colours. Our result is free from such error.

In Figure 4, 5 and 6, we compare our colourization results with that in [13] on images of different scenes and subjects. Generally speaking, our method obtains more visually appealing results, although the result is still affected by the choice of the training images (E.g., The colour tune in our squirrel image is warmer than the ground truth image, because it is so in the training image.). Note that the results of the child's image are not good. A possible reason is that the lack of small local features prevents matches. However, of the two results, ours is better.

5 Conclusion

In this paper, we have proposed a learning-based method of estimating the colours for a grayscale image. Compared with existing research on adding colours to monochrome images, our method is novel in that it extracts the input (grayscale) and the output (chromatic) samples as feature vectors from training data, views them as distributed in two manifolds with similar structures, and synthesizes colours for testing input image by exploring this similarity which is represented in reconstruction weights. Our method needs less user intervention than the previous work. It allows batch processing, because one training image set can be used for colourizing many images and no further interaction is needed.

In the future, we will adopt techniques in our framework to enhance the robustness of



(a) (b) (c) (d) (e)

Figure 4: Colourization of Squirrel Image
(a) training; (b) input; (c) ours; (d) as in [13]; (e) ground truth

the neighborhood searching. More sophisticated models, such as Markov random field, can also be adopted to handle the smoothness constraint on overlapped patches.

References

[1] D. Beymer and T. Poggio. Image representation for visual learning. *Science*, 1996.

[2] H. Chang, D.Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *Proceedings of CVPR*, 2004.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of CVPR*, 2005.

[4] David L. Donoho and Carrie Grimes. Image manifolds which are isometric to euclidean space. *J. Math. Imaging Vis.*, 23(1):5–24, 2005.

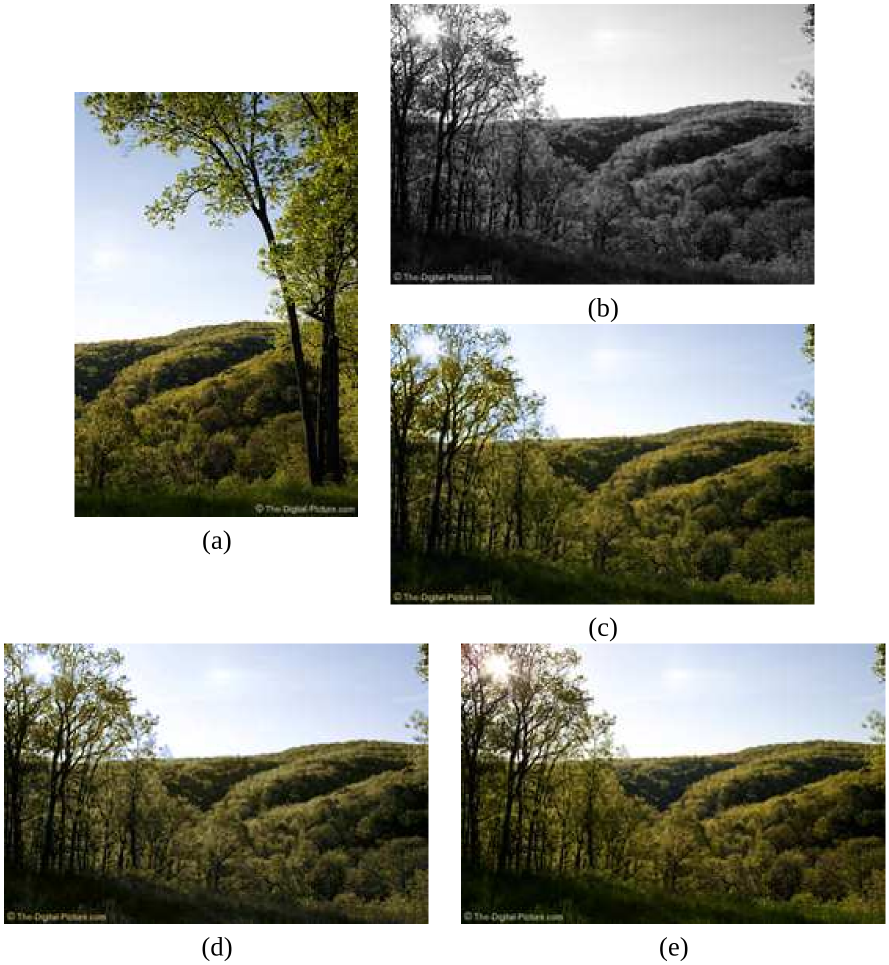


Figure 5: Colourization of Landscape Image II
(a) training; (b) input; (c) ours; (d) as in [13]; (e) ground truth

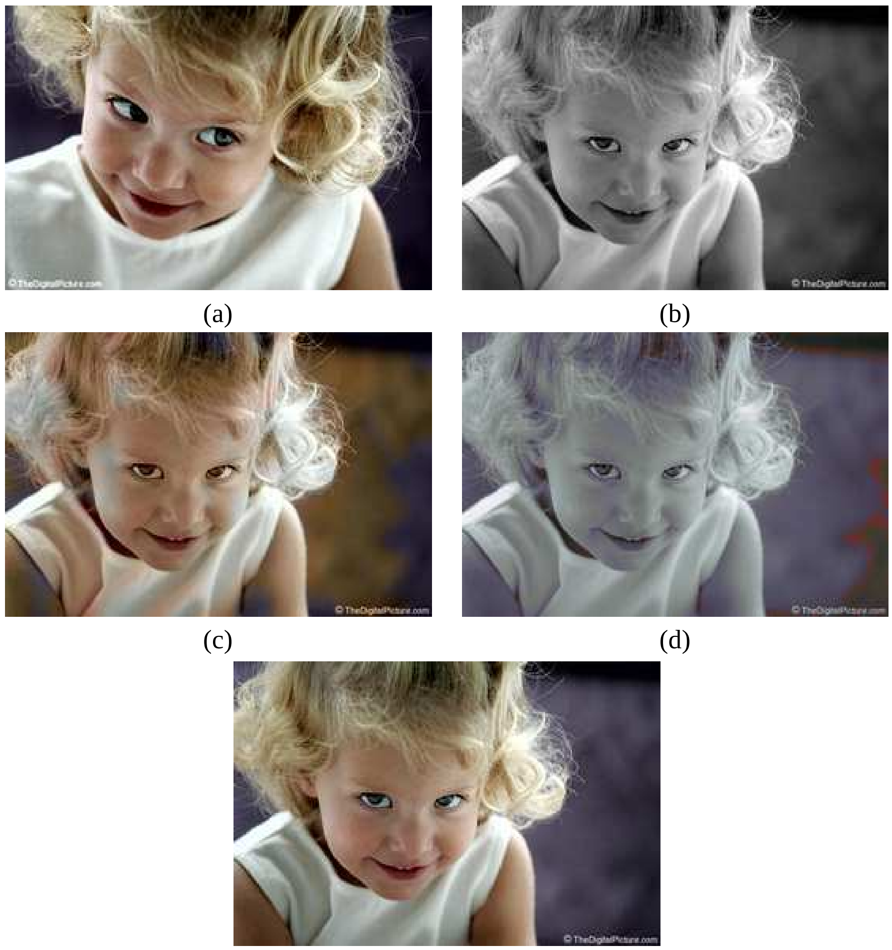
[5] Wei Fan and Dit-Yan Yeung. Image hallucination using neighbor embedding over visual primitive manifolds. In *Proceedings of CVPR*, 2007.

[6] X. Huo, X. Ni, and Andrew K. Smith. A survey of manifold-based learning methods. Technical report, Statistics Group, Georgia Institute of Technology, 2006.

[7] Yingge. Qu, Tien-Tsin Wong, and Pheng-Ann Heng. Manga colorization. In *In proceedings of Siggraph*, 2006.

[8] Sam T. Roweis and Lawrence Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 2000.

[9] H. Sebastian Seung and Daniel D. Lee. The manifold way of perception. *Science*, pages 2268–2269, 2000.



(e)

Figure 6: Colourization of Portrait Image
(a) train; (b) input; (c) ours; (d) as in [13]; (e) ground truth

[10] J. Silberg. Cinesite press article, 1998. http://www.cinesite.com/core/press/articles/1998/10_00_98-team.html.

[11] Richard. Souvenir. *Manifold learning for natural image sets*. PhD thesis, 2006.

[12] Jakob Verbeek. Learning non-linear image manifolds by combining local linear models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 28(8):1236–1250, 2006.

[13] Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller. Transferring color to greyscale images. In *In proceedings of Siggraph*, 2002.