Multiscale Texture Synthesis and Colourization of Greyscale Textures

Anders Hast
Creative Media Lab
University of Gävle
Kungsbäcksvägen 47
SE-801 76, Gävle, Sweden
aht@hig.se

Martin Ericsson
UPPMAX
Uppsala University
Lägerhyddsvägen 2
SE-751 05, Uppsala, Sweden
martin.ericsson@it.uu.se

Stefan Seipel
GraphiX Center
University of Gävle
Kungsbäcksvägen 47
SE-801 76, Gävle, Sweden
ssl@hig.se

ABSTRACT

The main idea presented herein is to use a multiscale texture synthesis approach in order to both colourize and upscale greyscale textures. Such textures can be vintage photos to be used in archaeological or urban 3D visualizations and obviously the colour needs to be reconstructed some way. Due to limited quality, walls etc in such 3D visualizations will appear either pixelized or blurry when the viewer approaches them on a close distance. The latter if some kind of interpolation technique is being used to reduce the pixelization. The low resolution greyscale texture and a high resolution coloured texture is used for the colourization and upscaling, which will produce a colour version of the greyscale texture with 4 times higher resolution in each upscale step. The novel idea is to use multiscale texture synthesis in HSV space for the first upscale in order to create a RGB colour image for subsequent upscaling, using either ordinary RGB multiscale texture synthesis or continue using HSV multiscale texture synthesis. These two main approaches will be compared and discussed.

Keywords

Multiscale Texture Synthesis, Colorization, Colour transfer, Greyscale Photos.

1. INTRODUCTION

In the process of 3D virtual reconstruction and visualization of buildings it is necessary to acquire textures of walls etc and often photographs are being used for obtaining the textures. However such walls and parts of buildings, especially for archaeological visualizations, might not exist anymore and it is therefore necessary to use old photos, often greyscale ones. There are hence two problems that must be solved. First of all the colour must be reconstructed, which might be a hard task unless we know at least something about the colours that were used when the building was still standing. Furthermore the quality must be improved because aliasing problems will occur when these textures are used for the 3D models. Usually different interpolation techniques are being used in order to minimize the aliasing effect when one moves close to the walls. One drawback with antialiasing [Fol97] is that it will make the texture look blurry, however this is preferred over having the pixels appear like big homogeneous square blocks, which makes the texture look pixelized.

We propose a novel approach for the colourization of a greyscale textures and a subsequent upscaling in order to increase the resolution so that the blurriness of interpolation can be avoided, using a modified variant of multiscale texture synthesis.

Multiscale Texture Synthesis

This paper does not deal with ordinary texture synthesis in general, but a short introduction will be given before we explain the basics of multiscale texture synthesis.

1.1.1 Texture Synthesis

Texture synthesis (TS) is the process of taking one smaller texture and then make it larger in size, not by tiling, but by synthesizing it [Efr99, Wei00]. Several approaches exist and the hierarchical TS method [Hee95] builds a tree of the texture with different sizes very much like in the mipmapping method [Wil83]. The smallest texture (on the lowest level) is then used in the first step and texels are randomly taken from it and randomly inserted into the new synthesized texture of corresponding size. Then follows a process where a mask with a specific shape [Har01] is scanning the synthesized texture in a scanline order fashion while copying texels from the original texture, which has the best matching neighbourhood [We02]. Generally the matching is computed as the sum of the squared differences of the RGB values within the masks. In the same time as the texture is synthesized on this level, another texture is synthesized on a higher level by copying a 2x2 neighbourhood into that texture, which accordingly will be 4 times larger.

1.1.2 Multiscale Texture Synthesis

In multiscale texture synthesis [Lee08] (MSTS) there already exists a texture version available of the otherwise initially randomized and then synthesized texture, namely the target texture. Then a number of examplar textures are taken so that they will contain similar details like the target texture but on higher levels, and they can subsequently be used to build a more detailed version of the target texture. An examplar graph is built for this purpose where the target texture is placed in the root and textures with higher details are placed on the next levels depending on their resolution. One texture on one level can thus depend on several exemplar textures on other levels. Since the colours can differ on different levels it has been proposed to use a colour transfer function [Han08]. Another approach can be used when the colours in the target texture are substantially different from the examplar texture [Has09], i.e. when the matching is bad.

1.1.3 HSV Multiscale Texture Synthesis

The HSV colour space MSTS method takes a different approach using only one examplar image [Has09]. As an example: when 3D reconstruction of buildings is used by the proposed approach it is possible to use an image of the whole wall or large parts of a wall (see figure 1) using a high resolution camera.

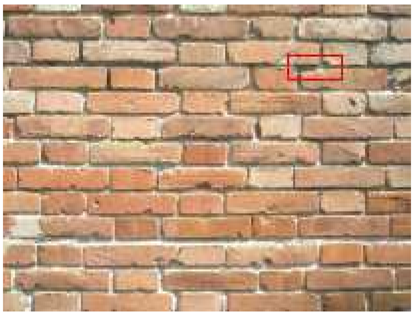

Figure 1. The target texture (204x153 pixels) with a red rectangle showing what part that will be zoomed.

The inserted details can be taken from an examplar image taken from the same wall. This image will be taken on a close range and will therefore cover a small part of the wall as shown in figure 2. Note that the texture to the left has been down sampled to fit in the paper. To the right is shown a small part, inside the red rectangle, in its actual resolution. It is obvious that the examplar texture has a high resolution compared to the target texture in figure 2, which is also shown in its actual resolution.

Figure 2. The examplar texture to the left, covering a smaller part of the wall. As shown to the right it has a high resolution (768x768 pixels) that will be used for the upscaling.

The HSV method for MSTS (HSV-MSTS) can handle colour differences in the following way: Let us say that a single brick in a brick wall have a greenish tone in the otherwise red wall. Then this problem can be handled by converting the colours into the Hue, Saturation and Value (HSV) colour space [Son99]. The HSV colour model separates the colour into three channels, similar to the more common red, green and blue colour model (RGB) but instead it uses a measurement of hue, saturation and value also called brightness. This model of representing colours gives the ability to change the brightness independent of the other colour information in the picture. As the human perceptive system is more sensitive to brightness discrepancies, this potentially can give a perceptually better image.

In figure 3 the original image to the left is compared to the resulting texture from ordinary MSTS in the middle and the image to the right shows the HSV-MSTS. It is obvious that the colours are not represented correctly in the middle image. Converting to HSV space and synthesizing the V part while interpolating H and S, will give a much more accurate result. The problematic greenish brick (inside the red truiangle) is synthesized keeping the greenish tone, using the HSV approach, while the ordinary MSTS makes it more red than green.

It should be noted that the target texture was originally larger in size and then down sampled by averaging 4 pixels into one. In this way it was possible to compare how close to the real thing the synthesis process was, i.e. we have a ground truth to compare with. Hence the texture to the right in figure 3 has a higher resolution than what was used in the process.

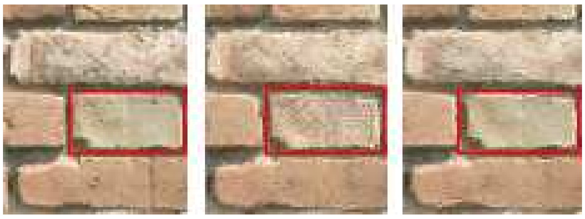

Figure 3. The high resolution target image to the left and a synthesised version in the middle using ordinary MSTS yielding colours, which are far from correct compared to the right where the HSV approach is used.

The synthesized textures on higher levels are constructed using the synthesized V elements and the H and S elements are taken from the target texture, which ensures that the original colours of the bricks are maintained. However it is important that the H and S elements are interpolated, e.g. bi-linearly (or using some other interpolation scheme [Gon93]), in the upscaling process, otherwise the colour will be visible as blocks.

Figure 4. Top: pixelization. Middle: Interpolation: Bottom: Multiscale Texture Synthesis (HSV).

In our approach a simple variant of bilinear interpolation was used taking into account only the 4-neigbours [Son99]. The result is shown in figure 4. In the top it can be seen how the texture (within the red rectangle in figure 1) is magnified without interpolation and in the middle interpolation has been used. Nonetheless the result is far from appealing. The result of the previously explained HSV-MSTS approach is shown in the bottom. It is obvious that this approach inserts details, making the image looking much better than just using interpolation in order to get rid of the pixelization.

Colourization of Greyscale Textures

Greyscale texture and image colourization is applied for an example in greyscale photo editing and scientific illustrations. The process of colourization increases the visual appeal of greyscale images and can perceptually enhance scientific illustrations [Che04]. It has also been used for colourization of classic movies, even though not all are that happy about that the visual experience is changed [Dan90].

Region based colourization can be performed by combining greyscale image matting algorithms [Smi96] with colour transferring techniques [Wel04]. First objects with that will have different colours are extracted from the greyscale image. Then each object is colourized using colour transferring and then these colourized objects are seamlessly composited [Mor95, Por84].

Colourization can be either user guided [Lev04] or automatic. And there are also techniques that use a combination of these two [Iro05]. The user guided method requires the user to scribble the desired colours in the interiors of the various regions. On the other hand, automatic techniques like the one proposed by Welsh et al [Wel02] colorizes an image by matching small pixel neighbourhoods in the image to those in the examplar image, and transferring colours accordingly. Hence they propose to use a variant of texture synthesis since they are matching local pixel luminance statistics between colour example and target grey-scale image.

The procedure according to Welsh et al [Wel02] and later used by Karthikeyani et al [Kar07] is as follows: first each image is converted into the $l\alpha\beta$-colour space [Rud98]. (Similarly Pan et al [Pan04] used this space to add colour to video and animation clips). Then jittered sampling is used to select a small subset of pixels in the colour image as samples. Next each texel in the greyscale image is traversed in scan-line order and the best matching is selected using neighbourhood statistics within in a 5x5 mask. The best match is determined by using a weighted average of texel luminance and the neighbourhood statistics. The chromaticity values (α,β channels) of the best match are then transferred to the greyscale. In order to obtain a better correspondence in the luminance range between the two images, luminance remapping [Her02] is performed.

2. COLOURIZING USING HSV-MSTS

This paper proposes how textures like wall textures can be colourized and upscaled using a novel idea that differs from the idea proposed mainly by Welsh et al [Wel02]. First of all we show that HSV-MSTS can be used for the colour transfer. Secondly we show that the upscaling process can be integrated in the process.

One of the reasons to use HSV-MSTS is that jittered samples would fail to find enough texels containing the mortar in the brick wall examples, unless the amount of samples is heavily increased. Furthermore we have found that the matching differences of the V value of each texel using a 3x3 mask is enough for a visually pleasing result, instead of a 5x5 mask matching neighbourhood statistics.

The novel idea is to use the previously explained fact that texture synthesis can be performed in HSV using the V channel for matching, in order to colourize a greyscale photo. In fact, for a greyscale photo only the V channel contains any information since there is no colour that can be saturated or be defined by its hue. Figure 5 shows three target textures in greyscale that will be colourized by the proposed approach. The examplar texture will be the same as shown in figure 2, that is downscaled to a size that corresponds to the size of the target textures. However we will do the matching with one difference, we will use a greyscale version of that texture for matching. Then we can proceed in two ways. When a best match is found we can take the corresponding pixel from the colour version of the examplar texture, either on the same level to construct a coloured version of the texture. As an alternative we can go a head and take the four corresponding pixels from the texture on a higher level (4 times larger), and hence make one step of upscaling on-the-fly.

Obtaining the Greyscale Examplar

Anyhow, the greyscale examplar image must be obtained from the colour version and this can be done in many ways. Since this paper presents a proof of concept rather than being applied on any certain vintage photo, we computed the greyscale examplar in the same way as we computed the target image as it was originally a colour image too.

Generally the greyscale value can be computed as a weighted sum of the RGB value:

$$g = r * w_r + g * w_g + b * w_b \quad (1)$$

Here we have the opportunity to arrange the weights so that the proposed algorithm works in the best way, i.e. the grey level histograms of the two images should be made as similar as possible.

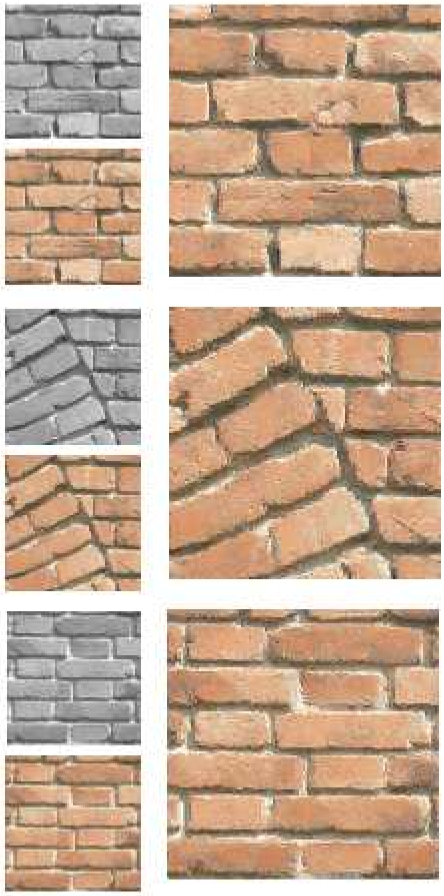

Figure 5. The three walls (top left) are colourized with no upscale (bottom left) and one step of 'on-the-fly' upscale to the right.

3. DISCUSSION

It should be noted that the target textures shown in figure 5 have a relatively low resolution, which is often the case for vintage photos. Nonetheless the algorithm works well despite the low resolution, but gives even better results if the resolution is higher. After the first colorization step we can proceed in another two ways, either we continue using the HSV variant or the RGB variant of MSTS.

In figure 6 the result from the subsequent upscalings are shown. Here we have been using the HSV approach all along. It appears like the bottom texture has a more spotty appearance compared to the more homogenously coloured texture in the top. Obviously the on-the-fly uspcaling introduces a noisy behaviour and it is better to upscale a colourized texture twice. Besides that, the textures are quite equal in appearance.

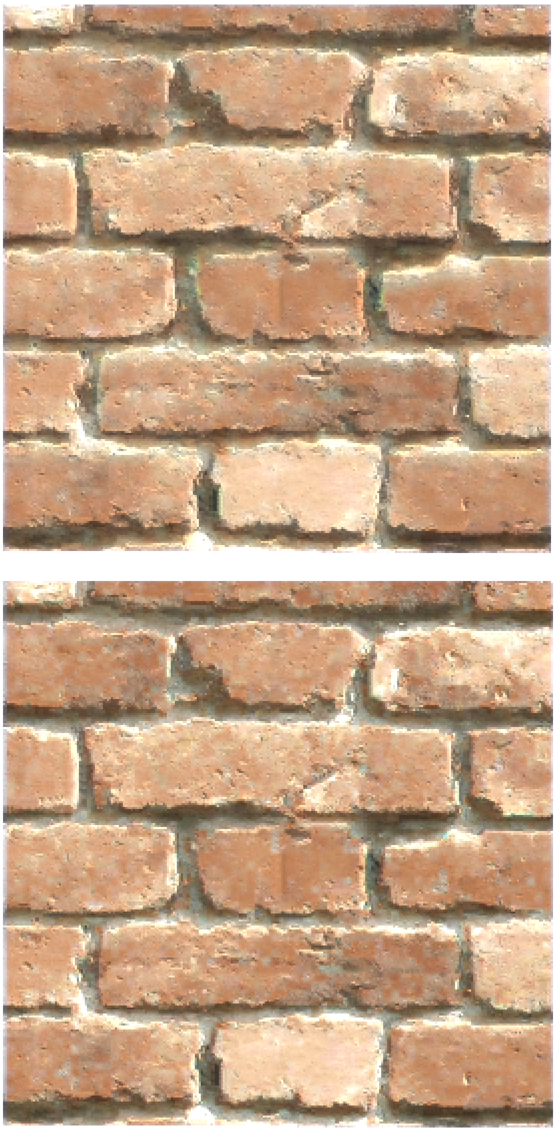

Figure 6. Top: the colourized wall HSV upscaled twice. Bottom: the colourized and on-the-fly HSV upscaled wall is then HSV upscaled once.

It should be noted that the V channel has been scaled with a small factor since the output was a bit darker compared to the original colour in the exemplar texture.

Next we go on to examine what happens if we would use the RGB variant for the subsequent steps and the result is shown in figure 7. It is quite hard to tell any difference in quality within the bricks, however the bottom images seem to have a less tendency to smear out details so that bricks become connected when there is just one or two pixels containing the mortar or when the border between bricks is quite fuzzy due to the low resolution. Therefore it seams like on-the-fly upscaling is to prefer when using the RGB-MSTS of the colorized textures but not for the HSV-MSTS.

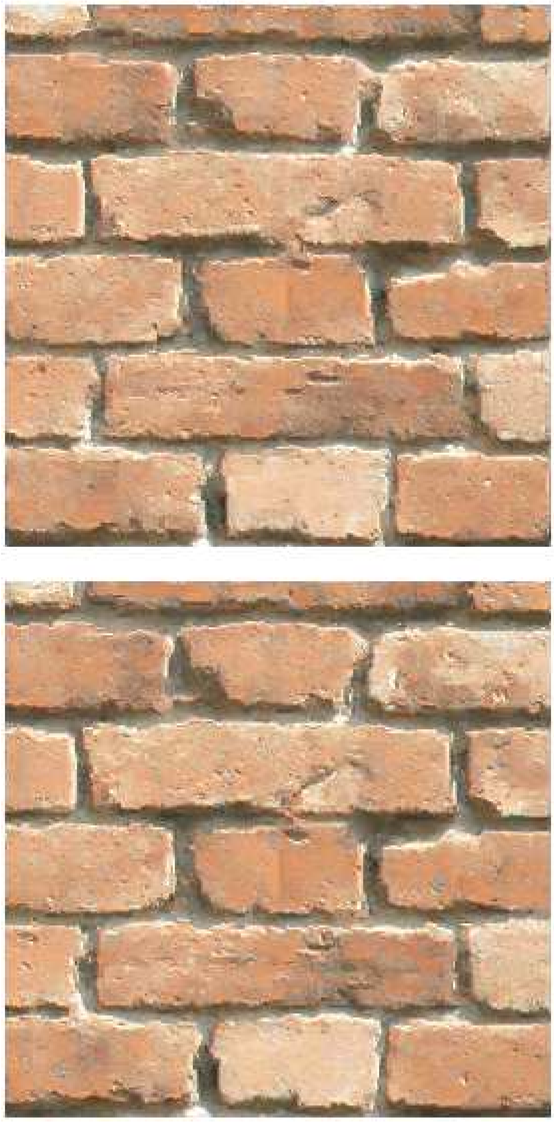

Figure 7. Top: the colourized wall upscaled twice. Bottom: the colourized and on the fly upscaled wall is upscaled once.

Furthermore it can be noticed that the smearing out does not really occur at all when using the HSV-MSTS all along.

A close up of another target texture showing the edge of a brick arch, that has been upscaled so it is 16 times larger is shown in figure 8.

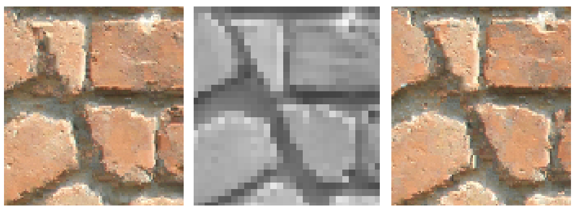

Figure 8. Left: the colourized wall upscaled twice. Middle: The original greyscale Right: The on-the-fly upscaled wall is upscaled once.

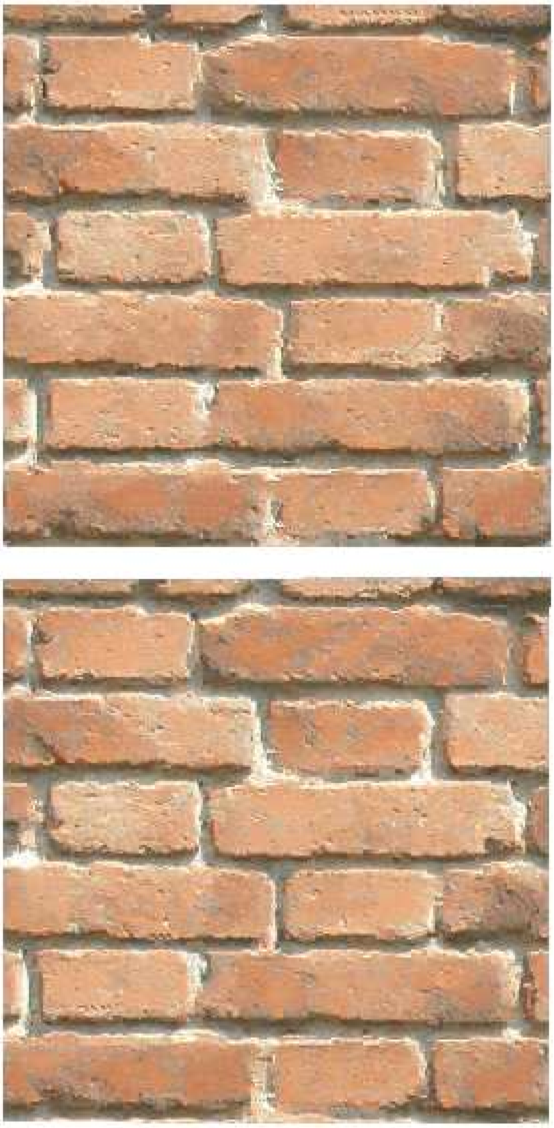

Figure 9. Top: the colourized wall upscaled twice. Bottom: the colourized and on the fly upscaled wall is upscaled once.

Figure 10. Top: the colourized wall upscaled twice. Bottom: the colourized and on the fly upscaled wall is upscaled once.

It is clear from the comparison that the on-the-fly upscaling generally gives a better result for greyscale textures with relatively low resolution as has been used in our tests. The same conclusion can be drawn from figure 9 and 10 where the other two test textures are shown (see also figure 5).

It should be remembered that even of the bottom textures are better they are not perfect and that is due to the extremely low resolution of the target textures used in our tests in order to test if the algorithm works for extreme cases. And therefore using higher spatial resolution will give even better results. It should also be noted that artefacts to great extent come from the fact that the matching was bad in these areas. This depends on too large differences in the image content between the target texture and the examplar. This could to great extent have been avoided using pre-processing of the images in order to normalize the intensities so that the impact of the flash etc is removed.

4. CONCLUSIONS

We have shown that colorization and upscaling of low resolution greyscale images can be performed using the recently published HSV-MSTS approach, using small masks like in our example a size of 3x3. Our tests also indicate that subsequent upscalings will become visually slightly better using RGB-MSTS and that on-the-fly upscaling is to prefer for this case.

Future Work

In the future we intend to develop the HSV-MSTS approach where one important task is to compare other colour spaces like the HSL and the lαβ-colour spaces. Another important task is to work on real vintage photos for colorization and upscaling.

5. REFERENCES

[Ale99] Alexei A. Efros and Thomas K. Leung. Texture Synthesis by Non-parametric Sampling, In Proceedings of ICCV 99, pp 1033-1038. 1999.

[Che04] T. Chen, Y. Wang, V. Schillings, and C. Meinel, Greyscale Image Matting and Colourization, Proceedings of Asian Conference on Computer Vision, pp. 1164-1169, 2004.

[Dan90] C. B. Daniels, Note on Colourization, The British Journal of Aesthetics, 30(1), pp68-70, 1990.

[Fol97] Foley, J. D., van Dam, A., Feiner, S. K., Hughes, J. F. Computer Graphics - Principles and Practice, Addison-Wesley, pp. 617-646. 1997.

[Gon93] Gonzales, R. C., Woods, R. E. Digital Image Processing, Addison-Wesley, pp. 300-302. 1993.

[Han08] Han, C., Risser, E., Ramamoorthi, R. and Grinspun, E., Multiscale Texture Synthesis. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2008), 27(3) pp. 51. 2008.

[Har01] P. Harrison, A non-hierarchical procedure for re-synthesis of complex textures. In proceedings of the WSCG'01, pp 190-197, 2001.

[Has09] A. Hast, M. Ericsson, T. Reiner, Improved Textures for 3D Virtual Reconstruction and Visualization by a Modified Multiscale Texture Synthesis Approach, ARCH-3D, pp. 1-6, 2009.

[Hee95] Heeger, D. and Bergen, J. Pyramid-Based Texture Analysis / Synthesis. SIGGRAPH pp. 229-238. 1995.

[Her02] A. Hertzmann, , C. Jacobs, N. Oliver, B. Curless, D. Salesin, 2001. Image Analogies, In Proceedings of ACM SIGGRAPH, pp. 341-346. 2002.

[Kar07] V. Karthikeyani, K.Duraiswamy, P. Kamalakkannan, Conversion of Grey-scale image to Color Image with and without Texture Synthesis IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.4, April 2007.

[Lee08] Lee, S-H., Park, H-W., Lee, J. and Kim, C-H. Multi-scale Texture Synthesis. Journal of KCGS (The Korea Computer Graphics Society), 14(2). 2008.

[Lev04] A. Levin, D. Lischinski, and Y. Weiss, *Colorization using Optimization*, Proceedings of ACM SIGGRAPH, pp. 689-694, 2004.

[Mor95] E. N. Mortensen and W. A. Barrett, Intelligent scissors for image composition, in Proceedings of ACM SIGGRAPH, pp. 191–198, 1995.

[Pan04] Z. Pan, Z. Dong, M. Zhang, A New Algorithm for Adding Color to Video or Animation Clips, WSCG, pp.515-519, 2004.

[Por84] T. Porter and T. Duff, Compositing digital images, in Proceedings of SIGGRAPH, pp. 253–259. 1984.

[Rud98] D. L. Rudermann., T. W. Cronin and C. C. Chiao, Statistics of Cone Responses to Natural Images: Implications for Visual Coding, J. Optical Soc. Of America, vol 15, no. 8, pp. 2036-2045, 1998.

[Wei00] Li-Yi Wei, Marc Levoy. Fast Texture Synthesis using Tree-structured Vector Quantization, Proceedings of Siggraph, pp 479-488. 2000.

[Wei02] Li-Yi Wei. Texture synthesis by fixed neighborhood searching. PhD Thesis, Stanford University, Palo Alto, CA, USA, 2002.

[Wel02] T. Welsh, M. Ashikhmin, and K. Mueller, Transferring colour to greyscale images," *ACM TOG*, vol. 20, no. 3, pp. 277–280, 2002.

[Wil83] Lance Williams, Pyramidal Parametrics, Computer Graphics, 17(3) pp.1-11. 1983

[Son99] Sonka, M., Hlavac, V. and Boyle, R. Image Processing, Analysis, and Machine Vision. Thomson Learning, USA. pp.28, 38. 1999.

[Smi96] A. R. Smith and J. F. Blinn, Blue screen matting," in Proceedings of ACM SIGGRAPH, pp. 259–268. 1996.