

Patch-Based Colour Transfer with Optimal Transport

Hana Alghamdi
 Computer Science & Statistics
 Trinity College Dublin
 Dublin, Ireland
 alghamdh@tcd.ie

Mairead Grogan
 V-SENSE, Computer Science & Statistics
 Trinity College Dublin
 Dublin, Ireland
 mgrogan@tcd.ie

Rozenn Dahyot
 Computer Science & Statistics
 Trinity College Dublin
 Dublin, Ireland
 Rozenn.Dahyot@tcd.ie

Abstract—This paper proposes a new colour transfer method with Optimal transport to transfer the colour of a source image to match the colour of a target image of the same scene. We propose to formulate the problem in higher dimensional spaces (than colour spaces) by encoding overlapping neighborhoods of pixels containing colour information as well as spatial information. Since several recoloured candidates are now generated for each pixel in the source image, we define an original procedure to efficiently merge these candidates which allows denoising and artifact removal as well as colour transfer. Experiments show quantitative and qualitative improvements over previous colour transfer methods. Our method can be applied to different contexts of colour transfer such as transferring colour between different camera models, camera settings, illumination conditions and colour retouch styles for photographs.

Index Terms—optimal transport, colour transfer, image enhancement, JPEG compression blocks.

I. INTRODUCTION

In the digital media post-production industry, colour transfer methods have been developed to transform a source colour image into a specified target colour image to match colour statistics or eliminate colour variations between different photographs. Colour variations between photographs often happen due to illumination changes, using different cameras, different in-camera settings or due to tonal adjustments of the users. Applications in image processing problems range from generating colour consistent image mosaicing and stitching [1] to colour enhancement and style manipulation [2].

When computing the transfer function, considering colour information only does not take into account the fact that coherent colours should be transferred to neighboring pixels, which can create results with blocky artifacts emphasizing JPEG compression blocks, or increase noise. In this paper we propose to extend the colour problem to higher dimensional spaces by encoding overlapping neighborhoods of pixels, taking into account both their colour and spatial information. We solve the high dimensional problem in 1D space using an iterative projection approach. Our method can be applied to different contexts of colour transfer such as transferring colour between different camera models, camera settings, illumination conditions and colour retouch styles for photographs.

This work is partly funded by a scholarship from Umm Al-Qura University, Saudi Arabia, and in part by a research grant from Science Foundation Ireland (SFI) under the Grant Number 15/RP/2776, and the ADAPT Centre for Digital Content Technology (www.adaptcentre.ie) that is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

II. RELATED WORKS

Optimal transport (OT) applications in imaging have been widely studied in the last decade. The OT problem consists of estimating the minimum cost (referred to as the Wasserstein Distance [3], [4] or as the Earth Movers Distance [5]) of transferring a source distribution to a target distribution. As a byproduct of OT distance estimation, the mapping ϕ itself between the two distributions is also provided.

Monge's formulation of OT [4] defines the deterministic coupling $y^c = \phi(x^c)$ between random vectors $x^c \sim f(x^c)$ and $y^c \sim g(y^c)$ that capture the colour information of the source and target images respectively, and its solution minimizes the total transportation cost:

$$\operatorname{argmin}_{\phi} \int \|x^c - \phi(x^c)\|^2 f(x^c) dx^c \quad (1.a)$$

$$\text{such that : } f(x^c) = g(\phi(x^c)) |\det \nabla \phi(x^c)| \quad (1.b)$$

with f the probability density function (pdf) of x^c and g the pdf of y^c . The solution for ϕ can be found using existing algorithms such as linear programming, and the Hungarian and Auction algorithms [6]. However, in practice it is difficult to find a solution for colour images when $\dim(x^c) = \dim(y^c) = d > 1$ as the computational complexity of these solvers increases in multidimensional spaces [3]. But for $d = 1$, with $x^c, y^c \in \mathbb{R}$, a solution for ϕ is straightforward and can be defined using the increasing rearrangement [4]:

$$\phi = G^{-1} \circ F \quad (2)$$

where F and G are the cumulative distributions of the colour values in the source and target images respectively. The 1D solution in (2) has been used to tackle problems in multidimensional colour spaces and of particular interest is the Iterative Distribution Transfer (IDT) algorithm for colour transfer proposed by Pitié et al. [7]. They proposed to iteratively project colour values $\{x_i^c\}_{i=1}^n$ and $\{y_j^c\}_{j=1}^m$ originally in \mathbb{R}^d to a 1D subspace and solve the OT using (2) in this 1D subspace and then propagate the solution back to \mathbb{R}^d space. This operation is repeated with different directions in 1D space until convergence. This strategy was inspired by the idea of the Radon Transform [8] which states the following proposition: if the target and source colour points are aligned in all possible 1D projective spaces, then matching is also achieved in \mathbb{R}^d

space. Note that the implementation of IDT approximates F and G using cumulative histograms¹.

The Sliced Wasserstein Distance (SWD) algorithm was later proposed, similarly using the iterative projection approach of IDT, and was applied to texture mixing applications [9] and colour transfer [10]. The SWD algorithm uses quantile matching instead of cumulative histogram matching to solve (2). More specifically, SWD sorts the n 1D projections of the source and target images respectively to define quantiles with regular increments of size $\frac{1}{n}$ between 0 and 1 for both source and target distributions. IDT can also be considered as an approximation of a quantile matching approach but with irregular increments derived from the cumulative histograms of the source and target images - as source and target quantiles do not match exactly, interpolation can be used to compute solution (2) [8]. The SWD algorithm can be computed in $\mathcal{O}(n \log(n))$ operations using a fast sorting algorithm [9]. When a small number of observations are available, using SWD is best but with a large number of observations, histogram matching with IDT is more efficient.

The warping map ϕ derived using the closed form (2) has a number of important characteristics: it is parameter free, no strong hypotheses are made about the distributions (i.e. f and g), and by design it is an increasing function. More importantly, it provides a tractable iterative solution for imaging applications formulated in higher dimensional spaces [10]. However, undesirable visual artifacts can be produced in the output image and a post processing step is added to ensure that the gradient field of the recoloured source image is as close as possible to the original source image [8]. Similarly, Bonneel et al. [10] use an iterative post-processing technique [11] to regularize the transportation map.

III. PATCH-BASED COLOUR TRANSFER

We propose to solve colour transfer by encoding overlapping neighborhoods of pixels, accounting for both colour information as well as pixel location, and extending the problem to higher dimensional space. We solve the transfer problem in 1D space using an iterative projection approach. In our context, we found that using the quantile matching approach that employs sorting operations is better than the histogram technique. The problem with building histograms is that the regular bin size used for every projection must be chosen, and so to alleviate the need to optimize the bin size parameter we choose the sorting technique [10]. Due to overlapping regions in patches, several recoloured candidates are now generated for each pixel in the source image. Therefore, we also define an original procedure to efficiently merge these candidates. Our method consists of the following steps:

1. Combine colour and spatial information. Zhixin et al. [12] have proposed to incorporate spatial information for portrait relighting that transfers shading with awareness of the face geometry in source and target images. Similarly we propose to incorporate pixel positions with colour information

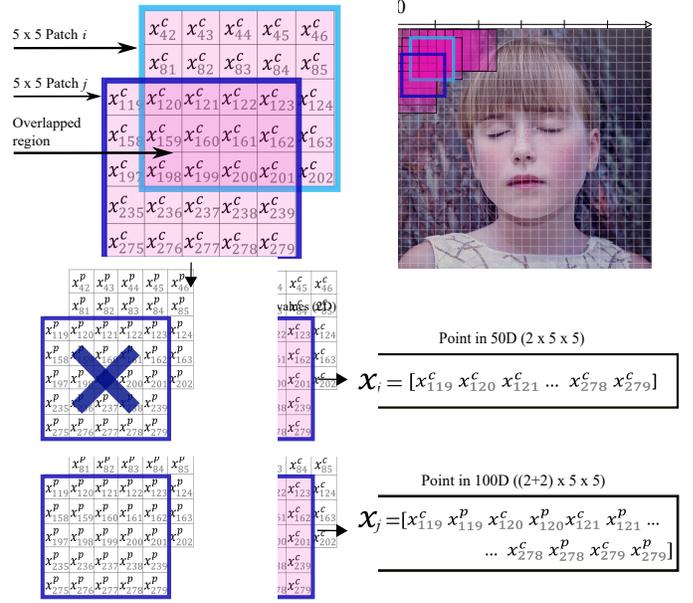


Fig. 1: Overlapping patches to create points in higher dimensions (best viewed in colour and zoomed in).

to take into account spatial information when recolouring same content images. The pixel's colour x^c and its pixel position x^p are concatenated into a vector $x = (x^c, x^p)^T$ such that $\dim(x) = \dim(x^c) + \dim(x^p)$. Adding this spatial information extends Monge's formulation in (1.a) such that $\phi(x)$ minimizes the following cost:

$$\int (\|x^c - \phi_c(x)\|^2 + \|x^p - \phi_p(x)\|^2) f(x) dx \quad (3.a)$$

$$\text{such that : } f(x) = g(\phi(x)) |\det \nabla \phi(x)| \quad (3.b)$$

where now f denotes the pdf of x and g the pdf of y in the high dimensional space (\mathbb{R}^d , where $d = \dim(x)$). ϕ_c and ϕ_p are the components of ϕ that return the transformed colour and spatial features respectively.

2. Change the coordinate system. In the case when we combine colour and spatial information, the colours have integer values from 0 to 255, and the spatial values can be anything depending on the size of the image. In order to produce consistent results regardless of the size of the image, we normalize all the colour and position coordinates to lie between 0 and 255 to create a hypercube in \mathbb{R}^d . We then stretch that space in the direction of the spatial coordinates by a factor $w = 2.5$ to make it harder to move the pixels in the spatial domain than in the colour domain, because since we are focusing on transferring colour between images of a same scene, we know that the scenes are overlapped.

3. Create patch vectors. We encode overlapping neighborhoods of pixels to preserve local topology information. Starting from the origin of the coordinate system of the images (upper left corner), we use a sliding window operation of window size $k \times k$ to extract overlapping patches. From each individual patch we create a high dimensional vector in $\mathbb{R}^{d \times k \times k}$ by concatenating the pixels in the patch with

¹<https://github.com/frcs/colour-transfer>

their d dimensional features (colour values only such that $x = (x^c)^T$ Fig. 1 (A), or colour and spatial features such that $x = (x^c, x^p)^T$ Fig. 1 (B)). We apply this process to the source and target images to create patch vector sets $\{x_i\}$ and $\{y_j\}$ for each respectively. Note, the formulation that includes spatial information favors local mapping between patches such that it penalizes mappings between patches that are spatially distant from each other even if their colours are similar.

4. Iterative projection transfer. The sorting operation from the SWD algorithm [10] is used as follows: let $x(t)$ represents the high dimensional source point in $\mathbb{R}^{d \times k \times k}$ at iteration t . At each iteration, we generate a random orthogonal basis for the $\mathbb{R}^{d \times k \times k}$ space and project the source and target points onto these axes. Along each axis, ϕ is estimated and then applied to the source points to create an intermediate value $\hat{x}(t+1)$. In SWD the final modified source points at the $t+1$ iteration are obtained by taking a partial step with $\alpha=0.2$ such that $x(t+1) = (1-\alpha)x(t) + \alpha\hat{x}(t+1)$. We have chosen here $\alpha=1$ to get an updating step similar to Pitié et al [7], hence not enforcing any pulling constraint on the transfer function towards the identity as Bonneel et al. [10]. After convergence, we retain only the transformed colour information (the $\phi_c(x)$ component of ϕ) and discard any changes in positions ($\phi_p(x)$).

5. Merge recoloured candidates. Because of the overlapping regions between patches, several recoloured candidates are now generated for each pixel in the source image. We compute the average colour value from the candidates and use it in conjunction with the original position to create the output recoloured image.

IV. EXPERIMENTAL ASSESSMENT

We provide quantitative and qualitative evaluations to validate both of our optimal transport solutions - using colour patches only, annotated in the results as *Our_c*, and using colour patches with spatial information, annotated as *Our_cp*. We compare our methods to different state of the art colour transfer methods [2], [8], [13], [14]. In these evaluations we use image pairs with similar content from an existing dataset provided by Hwang et al.² [2]. The dataset includes registered pairs of images (source and target) taken with different cameras and settings, and different illuminations and recolouring styles.

A. Colour space and parameters settings

We use the YCbCr colour space because of its ability to decouple colour channels. We transform the luminance (Y) and chrominance (CbCr) components separately and recombine the resulting recoloured sources (Y and CbCr) to create the final recoloured source image. For *Our_cp*, when processing chrominance, each pixel is represented by its 2D chrominance (CbCr) value and its 2D spatial position. When processing luminance, each pixel is represented by its 1D luminance (Y) values and its 2D spatial position. The spatial information

corresponds to the position coordinates of the pixel in the image, with the origin of the coordinate system in the upper left corner (Fig. 1). Our patches with combined colour and spatial features create a vector in 75 dimensions ($3 \times 5 \times 5$) for the luminance and position component, and 100 dimensions ($4 \times 5 \times 5$) for the chrominance and position component. For *Our_c*, pixel position is not accounted for, and only chrominance and luminance are used. When creating patch vectors, we experimented with different patch sizes and found that a size of 5×5 captures enough of a pixel's neighbourhood and does not increase computational complexity too much. The code we used to implement our method is available online³.

B. Evaluation metrics

To quantitatively assess the recolouring results, four metrics are used: peak signal to noise ratio (PSNR) [15], structural similarity index (SSIM) [16], colour image difference (CID) [17] and feature similarity index (FSIMc) [18]. These metrics are often used when considering source and target images of the same content [2], [14], [19], [20]. Although the registration errors may affect the evaluation metrics, the measures for different methods were computed using the same registration errors. Note that the results using PMLS were provided by the authors [2]. We also compared PMLS with two other recent techniques [21], [22] that also incorporate correspondences into their framework, but PMLS has been shown to perform better than these two [13] and so is the one reported here.

C. Experimental Results

Table I presents the quantitative results. Our method with colour and spatial information (*Our_cp*) outperforms other state of the art methods in most cases as measured by CID and FSIMc and performs similarly to other top methods PMLS and L2 as measured by the PSNR and SSIM metrics.

Figure 2 provides qualitative results. For clarity, the results are presented in image mosaics, created by switching between the target image and the transformed source image column wise (Figure 2, top row). If the colour transfer is accurate, the resulting mosaic should look like a single image (ignoring the small motion displacement between source and target images), otherwise column differences appear. As can be noted, our approach *Our_cp* is visually the best at removing the column differences.

While PMLS provides equivalent results to our methods in terms of metrics, it introduces visual artifacts if the input images are not registered correctly (Figure 3), while our method is robust to registration errors. Note that although the accuracy of the PSNR, SSIM, CID and FSIMc metrics relies on the fact that the input images are registered correctly; if this is not the case, these metrics may not accurately capture all artifacts (Figures 3 and 2). In addition, due to the merging step of our algorithm (cf. step 5 Sec. III), our approach allows us to create a smooth colour transfer result, and can also alleviate JPEG compression artifacts and noise (cf. Figure 3 for comparison).

²<https://sites.google.com/site/unimono/pmls>

³https://github.com/leshep/PCT_OT

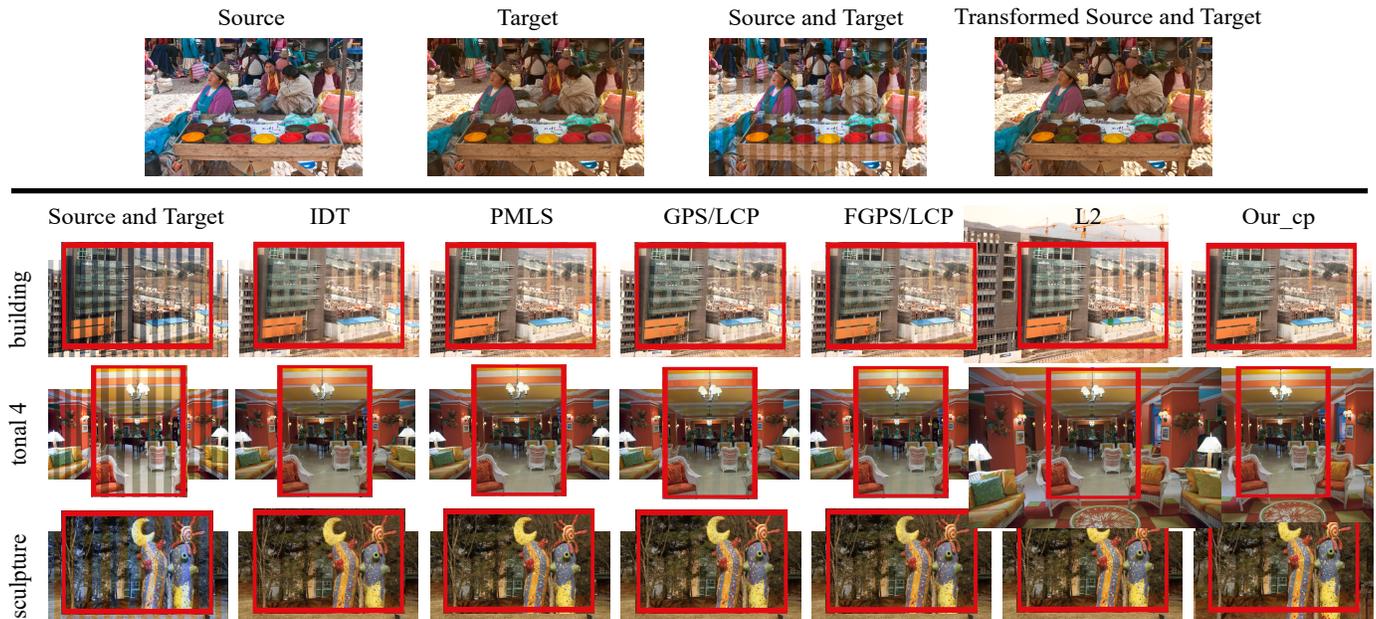


Fig. 2: Qualitative evaluations (best viewed in colour and zoomed in). The top result is processed with our technique *Our_cp*.

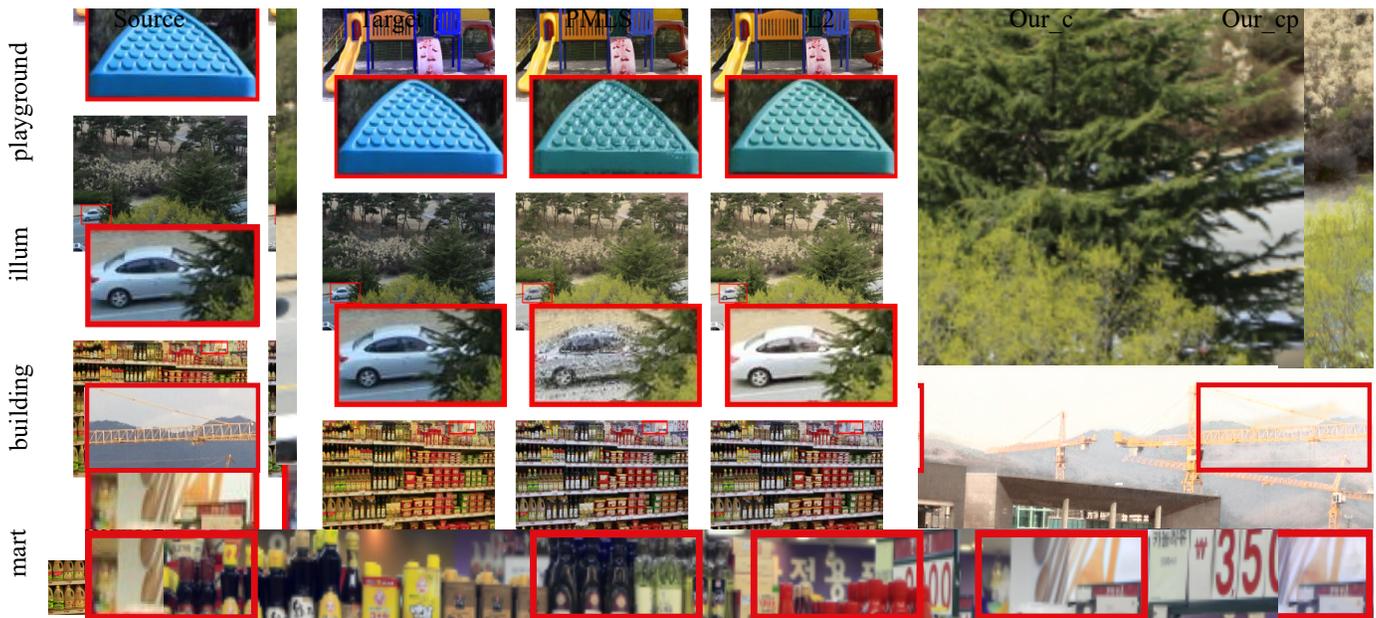


Fig. 3: A close up look at some of the results generated using the PMLS, L2 and our algorithms (best viewed in colour and zoomed in).

Our method *Our_cp* can also correctly transfer colours between images that contain moving objects, as can be seen in Figure 3 with examples ‘illum’ and ‘mart’. However, *Our_cp* can create shadow artifacts when there are large changes between target and source images (Figure 3 in example ‘building’). In this case *Our_c* does not suffer from these shadow artifacts and creates good colour transfer results. In the future, we would like to create a hybrid framework that integrates both *Our_c* and *Our_cp* and switches between the two using the parameter w . A motion estimation technique can be used to determine which pixels in the source image have moved in the target image. With pixels affected by large motion, the position’s weight would be set to $w = 0$, making

the algorithm account for *Our_c* only. When pixels move less, the position’s weight would be set to $w = 2.5$, making the algorithm account for *Our_cp*.

V. CONCLUSION

Several contributions to colour transfer within OT have been made in this paper, showing quantitative and qualitative improvements over state of the art methods. In particular, first, neighborhoods of pixels (patches) are used with OT algorithm in high dimensional space, and second, spatial information as well as colour content of pixels are both encoded in the high dimensional feature vectors. This original construction implies a new reconstruction step since each recoloured pixel benefits from the contribution of several estimated candidates using a

	PSNR					SSIM					CID					FSIMc												
	IDT	PMLS	GPS/LCP	FGPS/LCP	L2	Ours_c	Ours_sp	IDT	PMLS	GPS/LCP	FGPS/LCP	L2	Ours_c	Ours_sp	IDT	PMLS	GPS/LCP	FGPS/LCP	L2	Ours_c	Ours_sp							
gangnam1	25.354	35.725	24.048	23.936	35.358	25.217	31.479	0.900	0.992	0.892	0.891	0.990	0.920	0.964	0.252	0.040	0.226	0.222	0.048	0.189	0.085	0.936	0.986	0.943	0.985	0.946	0.972	
gangnam2	27.116	36.553	25.952	25.944	35.524	28.300	35.502	0.920	0.993	0.909	0.909	0.986	0.938	0.980	0.268	0.039	0.291	0.292	0.089	0.217	0.068	0.952	0.962	0.962	0.988	0.968	0.990	
gangnam3	22.372	35.007	21.908	21.913	33.284	22.373	26.393	0.859	0.991	0.873	0.864	0.980	0.849	0.930	0.496	0.108	0.487	0.487	0.193	0.484	0.261	0.946	0.962	0.961	0.990	0.973	0.987	
illum	19.822	20.167	19.785	19.960	19.079	19.370	20.306	0.641	0.649	0.643	0.646	0.648	0.657	0.673	0.386	0.390	0.395	0.396	0.397	0.428	0.377	0.800	0.819	0.824	0.824	0.818	0.839	
building	20.554	22.634	22.736	22.769	20.499	20.761	25.019	0.808	0.865	0.864	0.863	0.862	0.847	0.888	0.374	0.228	0.313	0.321	0.249	0.313	0.183	0.874	0.928	0.930	0.929	0.926	0.912	0.942
playground	27.184	27.835	25.501	25.436	27.647	26.002	28.482	0.920	0.940	0.878	0.876	0.939	0.902	0.939	0.440	0.238	0.443	0.471	0.254	0.466	0.209	0.950	0.958	0.933	0.932	0.955	0.938	0.956
flower1	24.238	26.981	23.765	23.706	26.857	21.661	25.186	0.909	0.967	0.913	0.912	0.966	0.874	0.926	0.389	0.163	0.396	0.400	0.174	0.458	0.285	0.954	0.975	0.968	0.967	0.976	0.971	
flower2	25.417	25.760	25.259	25.223	25.772	24.519	26.373	0.901	0.928	0.894	0.894	0.927	0.893	0.933	0.337	0.245	0.322	0.323	0.266	0.363	0.218	0.941	0.950	0.945	0.945	0.949	0.954	
tonal1	30.082	37.215	31.617	31.413	37.332	31.451	37.044	0.953	0.988	0.971	0.970	0.987	0.966	0.988	0.310	0.101	0.285	0.308	0.111	0.294	0.097	0.964	0.997	0.986	0.986	0.997	0.993	0.998
tonal2	27.992	31.508	25.062	25.087	31.356	28.376	32.049	0.968	0.987	0.926	0.926	0.986	0.971	0.988	0.288	0.128	0.351	0.347	0.145	0.250	0.099	0.984	0.993	0.973	0.973	0.992	0.984	0.993
tonal3	29.575	36.246	28.136	28.065	36.644	29.704	33.793	0.962	0.992	0.947	0.946	0.992	0.968	0.987	0.244	0.079	0.294	0.294	0.081	0.206	0.077	0.979	0.997	0.984	0.983	0.997	0.993	0.997
tonal4	28.605	34.521	28.852	28.848	34.344	28.209	33.819	0.944	0.983	0.932	0.932	0.983	0.948	0.981	0.240	0.108	0.248	0.238	0.107	0.173	0.065	0.966	0.989	0.972	0.973	0.990	0.987	0.995
tonal5	30.330	35.260	29.580	29.448	34.303	31.321	36.437	0.965	0.986	0.953	0.954	0.985	0.972	0.990	0.156	0.091	0.205	0.192	0.092	0.195	0.051	0.980	0.994	0.987	0.987	0.993	0.991	0.998
mart	22.747	24.742	23.183	23.196	24.450	22.890	24.509	0.904	0.957	0.925	0.925	0.956	0.918	0.941	0.526	0.219	0.405	0.402	0.225	0.526	0.426	0.946	0.969	0.960	0.959	0.967	0.957	0.969
sculpture	29.884	32.062	29.037	28.820	32.067	26.402	31.237	0.942	0.971	0.934	0.932	0.972	0.885	0.945	0.137	0.137	0.213	0.224	0.143	0.381	0.232	0.980	0.987	0.982	0.980	0.988	0.982	0.988

TABLE I: Quantitative evaluations. Red, blue, and green indicate 1^{st} , 2^{nd} , and 3^{rd} best performance respectively when applied to dataset [2] that includes captured image pairs with different cameras, camera setups, illumination and different image recolouring styles (best viewed in colour and zoomed in).

simple averaging solution which allows denoising and artifact removal as well as colour transfer.

REFERENCES

- [1] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International Journal of Computer Vision*, vol. 74, no. 1, pp. 59–73, Aug 2007. [Online]. Available: <https://doi.org/10.1007/s11263-006-0002-3>
- [2] Y. Hwang, J. Lee, I. S. Kweon, and S. J. Kim, "Color transfer using probabilistic moving least squares," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2014, pp. 3342–3349.
- [3] C. Villani, *Topics in optimal transportation*. American Mathematical Soc., 2003, no. 58.
- [4] —, *Optimal transport: old and new*. Springer Science & Business Media, 2008, vol. 338.
- [5] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, Nov 2000. [Online]. Available: <https://doi.org/10.1023/A:1026543900054>
- [6] F. Santambrogio, "Optimal transport for applied mathematicians," *Birkhäuser, NY*, pp. 99–102, 2015.
- [7] F. Pitié, A. C. Kokaram, and R. Dahyot, "N-dimensional probability density function transfer and its application to color transfer," in *IEEE Int. Conf. on Computer Vision (ICCV)*, vol. 2, Oct 2005, pp. 1434–1439 Vol. 2.
- [8] —, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1, pp. 123 – 137, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1077314206002189>
- [9] J. Rabin, G. Peyré, J. Delon, and M. Bernot, "Wasserstein barycenter and its application to texture mixing," in *Scale Space and Variational Methods in Computer Vision*. Springer Berlin Heidelberg, 2012, pp. 435–446.
- [10] N. Bonneel, J. Rabin, G. Peyré, and H. Pfister, "Sliced and radon wasserstein barycenters of measures," *Journal of Mathematical Imaging and Vision*, vol. 51, no. 1, pp. 22–45, Jan 2015. [Online]. Available: <https://doi.org/10.1007/s10851-014-0506-3>
- [11] J. Rabin, J. Delon, and Y. Gousseau, "Removing artefacts from color and contrast modifications," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3073–3085, Nov 2011.
- [12] Z. Shu, S. Hadap, E. Shechtman, K. Sunkavalli, S. Paris, and D. Samaras, "Portrait lighting transfer using a mass transport approach," *ACM Trans. Graph.*, vol. 37, no. 1, Oct. 2017. [Online]. Available: <http://doi.acm.org/10.1145/3095816>
- [13] M. Grogan and R. Dahyot, "L2 divergence for robust colour transfer," *Computer Vision and Image Understanding (in press)*, 2019.
- [14] F. Bellavia and C. Colombo, "Dissecting and reassembling color correction algorithms for image stitching," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 735–748, Feb 2018.
- [15] D. Salomon, *Data compression: the complete reference*. Springer Science & Business Media, 2004.
- [16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, April 2004.
- [17] J. Preiss, F. Fernandes, and P. Urban, "Color-image quality assessment: From prediction to optimization," *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1366–1378, March 2014.
- [18] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "Fsim: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug 2011.
- [19] I. Lissner, J. Preiss, P. Urban, M. S. Lichtenauer, and P. Zolliker, "Image-difference prediction: From grayscale to color," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 435–446, Feb 2013.
- [20] M. Oliveira, A. D. Sappa, and V. Santos, "A probabilistic approach for color correction in image mosaicking applications," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 508–523, Feb 2015.
- [21] J. Park, Y. Tai, S. N. Sinha, and I. S. Kweon, "Efficient and robust color consistency for community photo collections," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 430–438.
- [22] M. Xia, J. Y. Renping, X. M. Zhang, and J. Xiao, "Color consistency correction based on remapping optimization for image stitching," in *IEEE Int. Conf. on Computer Vision Workshops (ICCVW)*, Oct 2017, pp. 2977–2984.