

# Local Signal Equalization for Correspondence Matching

Derek Bradley  
Disney Research Zurich

Thabo Beeler  
Disney Research Zurich

## Abstract

*Correspondence matching is one of the most common problems in computer vision, and it is often solved using photo-consistency of local regions. These approaches typically assume that the frequency content in the local region is consistent in the image pair, such that matching is performed on similar signals. However, in many practical situations this is not the case, for example with low depth of field cameras a scene point may be out of focus in one view and in-focus in the other, causing a mismatch of frequency signals. Furthermore, this mismatch can vary spatially over the entire image. In this paper we propose a local signal equalization approach for correspondence matching. Using a measure of local image frequency, we equalize local signals using an efficient scale-space image representation such that their frequency contents are optimally suited for matching. Our approach allows better correspondence matching, which we demonstrate with a number of stereo reconstruction examples on synthetic and real datasets.*

## 1. Introduction

Computing correspondences between images is a fundamental problem in computer vision. Correspondence matching is key to many applications including stereo and multi-view reconstruction, optical flow, template matching, image registration and video stabilization (to name just a few). In this work we focus on patch-based correspondence matching, often used for dense stereo reconstruction.

Several metrics have been developed to measure the similarity of image patches. One of the most common measures is normalized cross-correlation (NCC), which forms the basis for a wide range of matching algorithms. The drawback of NCC is that it assumes the input images contain the same signal content, either all images are true representations of the original scene signal or all are equally distorted through defocus, motion blur, under-sampling, etc. However, in many practical situations we have to compute correspondence of mixed-signals, caused by different amounts of defocus blur for example. Furthermore, the signal degradation will vary spatially in most practical situations causing different mixed-signals for every pixel in an image.

In this paper, we take a pre-processing approach to solve the signal mismatch problem. Rather than defining new measures and devising new algorithms for matching mixed-signals, we believe it is more advantageous to *modify* the input signals in an optimal way to enable the wide range of existing algorithms which assume the signals are already equalized. To this end, we propose to locally equalize the image signals before computing correspondence matches. Given two images patches, our technique involves a frequency equalization step in order to generate patches with similar spatial signals, and an optimal frequency scaling step that ensures the patches can be matched reliably. This is achieved using a notion of local image frequency, which quantifies the highest spatial frequency in a local region, and an efficient implementation of scale-space that builds multi-resolution image patches on demand. The result is a pair of image patches that contain similar representations of the real scene, with sufficient local details for accurately computing cross-correlation.

We demonstrate the power of the proposed signal equalization technique by improving the performance of a naïve window matching algorithm by an order of magnitude on several examples of dense stereo reconstruction, using a synthetic data set with ground truth, and additional real-world examples. While we focus on stereo, our method can have significant impact on several other image processing and computer vision applications that rely on patch-based correspondence matching. In addition, our method is a general pre-processing technique that can benefit a wide range of existing algorithms.

## 2. Related Work

Correspondence matching is very common in several domains, including stereo reconstruction [20], multi-view reconstruction [21], optical flow [1], and many more. In this work we address the general issue of inconsistent spatial frequency signals between multiple images when performing correspondence estimation. While there exist many patch-based correlation measures [10] for stereo, they typically assume that frequency signals are consistent in the images before matching. We present a technique for establishing consistent frequency signals automatically.

A mismatch of signals between images can occur due to differently (de-)focused cameras. While some approaches have been developed for recovering depth from defocus information [6, 23, 12], we have a different goal, which is to improve correspondence matching between multiple images containing different signals. Goesele et al. [8] avoid the issue by collecting a large set of images from the internet and then choosing ones that have similar properties, however this option is only viable for reconstructing well-known and well-photographed landmarks.

In correspondence matching, it is well-known that signal content is related to the matching window size, and previous work has adapted the window to the image content using disparity uncertainty measures [13], edge information [16], entropy [5], object boundaries [24], correlation [18], intensity variation [9], or perspective distortion [19, 3]. However, these approaches do not solve the "mixed" signal problem, and in contrast we aim to explicitly locally *equalize* the signal of both images before computing correspondences.

Signal analysis in stereo reconstruction is not an entirely new concept. Klowsky et al. [14] analyze signal content in patch-based stereo systems, showing that the amplitude of high-frequency details is diminished in reconstructions, and they derive a modulation transfer function in frequency space that can be inverted to restore details. Following this theory, they show how Gaussian weighting of stereo patch correspondences links reconstruction to a scale-space representation of the underlying surface [15]. While this work is related, it is orthogonal to our own. Strecha et al. [22] learn ways of compressing feature descriptors and improve descriptor invariance under a certain class of degradations. Our method is complementary to this work, since our signal equalization approach can be applied before any type of descriptor matching or learning techniques.

### 3. Theoretical Considerations

We begin with a theoretical discussion of signal equalization. We aim to equalize two signals  $G_1(\omega)$  and  $G_2(\omega)$ , which are degraded versions of the same signal  $\hat{G}(\omega)$ , such that their frequency contents are both compatible and optimally suited for matching with respect to a given matching function  $\Psi$ . In this paper we focus on the normalized cross-correlation  $\Psi_{NCC}$ , one of the most widely used matching functions, which is defined as:

$$\Psi_{NCC}[g, f](t) := \frac{\int_{-\infty}^{\infty} g(\tau)f(\tau+t)d\tau}{\sqrt{\int_{-\infty}^{\infty} g(\tau)^2d\tau \int_{-\infty}^{\infty} f(\tau+t)^2d\tau}} \quad (1)$$

for two real valued functions  $g$  and  $f$ . We first discuss the equalization of two different signals, and then a theory of frequency scaling to produce optimal conditions for correspondence matching.

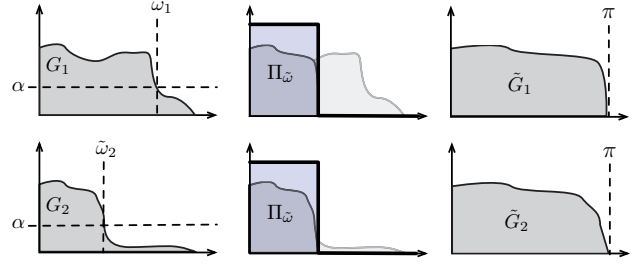


Figure 1. **Signal Equalization** - Two degraded versions ( $G_1, G_2$ ) of the same signal can be equalized. *Left*: A threshold  $\alpha$  is applied in frequency domain to find the highest frequencies  $\tilde{\omega}_1$  and  $\tilde{\omega}_2$ . *Center*: The lower of the two is used as cutoff frequency  $\tilde{\omega}$  for a lowpass filter  $\Pi_{\tilde{\omega}}$ . *Right*: Scaling the frequencies by  $\pi/\tilde{\omega}$  spreads them over the available domain, which is ideal for matching.

#### 3.1. Inter-Signal Equalization

In the frequency domain, the degradation is modeled as

$$G(\omega) = D(\omega)\hat{G}(\omega) + \mu(\omega), \quad (2)$$

where  $D(\omega)$  is the degradation function and  $\mu(\omega)$  is the noise model. Since the degradation we are primarily interested in is spatial blur due to de-focus, which corresponds to low-pass filtering  $\hat{G}$ , we model  $D$  as box filter  $\Pi_{\tilde{\omega}}$ , where  $\tilde{\omega}$  is the cutoff frequency. The cutoff frequency is set to be the highest frequency present in the spectrum.

To be robust with respect to noise, we employ a threshold  $\alpha$  to find the highest frequencies  $\tilde{\omega}_1$  and  $\tilde{\omega}_2$  of the two signals  $G_1$  and  $G_2$ , respectively. The lower of the two is then used as cutoff frequency for the lowpass filter  $\tilde{\omega} = \min(\tilde{\omega}_1, \tilde{\omega}_2)$  (Fig. 1 (left)). Multiplying both signals in frequency domain by  $\Pi_{\tilde{\omega}}$  will render them compatible for matching (Fig. 1 (center)). However, the frequency content is not guaranteed to be optimal for the matching function  $\Psi_{NCC}$ , which we address in the next section.

#### 3.2. Frequency Scaling

We are interested in finding the frequency for which spatial localization is best. This is fulfilled when a deviation from the minimum at  $t = 0$  by a spatial differential amounts in a maximal change of  $\Psi_{NCC}$ . Considering only a single frequency  $\omega$  this can be formalized as

$$\left\| \frac{d^2}{dt^2} \Psi_{NCC}[\sin_\omega, \sin_\omega](t) \right\| = \left\| -\omega^2 \cos(\omega t) \right\|_{t=0} = \omega^2. \quad (3)$$

This confirms the intuition that the higher the spatial frequency  $\omega$  the better the signal can be localized. Due to the discrete nature of an image, the shortest wavelength possible is  $\lambda = 2$  pixels and thus the highest frequency is  $\omega_{max} = 2\pi\frac{1}{\lambda} = \pi$ . For optimal matching, the frequencies should therefore be scaled by  $\pi/\tilde{\omega}$  to span the full domain, such as illustrated in Fig. 1 (right).

### 3.3. Combined Signal Equalization

By combining the findings from sections 3.1 and 3.2 the fully equalized signal is given as

$$\tilde{G}(\omega) = \Pi_{\tilde{\omega}} G\left(\omega \frac{\pi}{\tilde{\omega}}\right). \quad (4)$$

## 4. Equalization Implementation

In this section we will describe our implementation of local signal equalization for correspondence matching. Following the theoretical considerations introduced in Section 3, the algorithm requires to identify the joint cutoff frequency  $\tilde{\omega}$  to equalize the signals and scale the signal frequencies. We first define a method for computing local frequency information, which forms the basis to robustly identify the cutoff frequency. We then address frequency scaling using a scale-space approach, which is implemented using a locally centered pyramid structure. Finally, these tools are used to perform local signal equalization before correspondence matching, for example in stereo reconstruction.

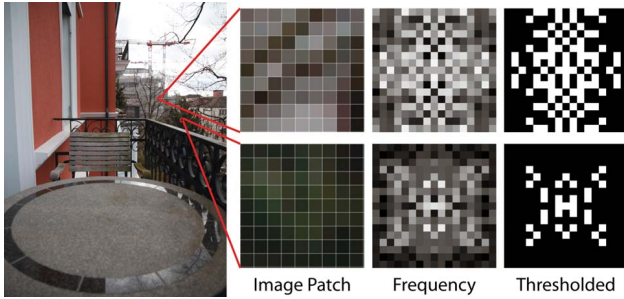


Figure 2. **Local Frequency** - Frequency content illustrated for two patches of an image. The discrete Fourier transform is computed for each patch and amplitudes are thresholded.

### 4.1. Local Frequency Maps

As we discussed in Section 3, an image patch that contains high spatial frequencies matches more robustly than one with only low frequency information, and the best matching frequency is the highest possible. Motivated by this theory, we aim to compute *local* frequency information at each point in an image and analyze the highest spatial frequency within the local region.

We employ the Discrete Fourier Transform (DFT) to measure image frequency, and we individually compute the DFT of local patches in the image. The user-defined correspondence matching window of size  $k$  is also used as the region of interest for the DFT, ensuring that we compute exactly the frequencies that will be used when matching. Fig. 2 illustrates the local frequency at two different points in an image.

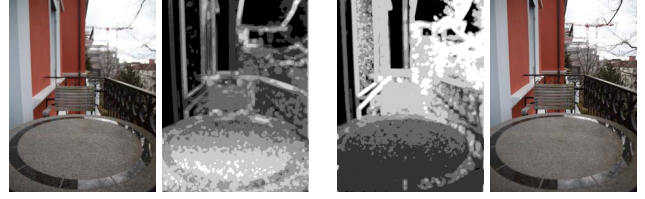


Figure 3. **Local Frequency Map** - Maps of two images with different focal planes ( $k = 9, \alpha = 0.4$ ). *Left*: focused near. *Right*: focused far.

In order to determine the highest reliable frequency in each local patch we need to account for noise, by thresholding on the frequency amplitudes. Let  $\Omega^p = \{\omega_i^p\}$  be the discrete set of frequencies for a patch  $p$  with corresponding amplitudes  $A^p = \{a_i^p\}$ . We then determine the highest frequency  $\tilde{\omega}^p$  as

$$\tilde{\omega}^p = \max_i(\Omega^p | a_i^p > \alpha), \quad (5)$$

where  $\alpha$  characterizes the amplitude of the image noise in the frequency domain. This parameter can be computed automatically if the noise is known, or it can be set empirically. The right column of Fig. 2 illustrates the thresholded frequency. Computing  $\tilde{\omega}^p$  for every pixel results in a *local frequency map*, which we illustrate for two images in Fig. 3. The images show the same scene but captured with different focal planes. Notice that the local frequency map is highly correlated with the image focus.

This measure of local highest frequency will guide us to equalize local signals for image correspondence matching.

### 4.2. Efficient Scale-Space Representation

In order to perform frequency scaling, the local signal at an image pixel can be altered by traversing the scale-space of the image. Specifically, changing levels in the scale space corresponds to frequency scaling of the original signal. The most common representation of a scale-space is the Gaussian image pyramid. A typical image pyramid contains the original image on the lowest level ( $\ell_0$  in our terminology), and increasingly filtered and downsampled versions at higher levels. Pyramids are most often used to perform computations at lower resolutions and then propagate the result as a prior for computations at higher resolutions. Stereo matching between two images is a classical example. However, this technique only works if the signal content is consistent between images at each level. If the signal is inconsistent on any level, then matching can become erroneous (see our experiments in Section 5). To overcome this issue, we determine the optimal scale for matching, in a scale-space centered on the matching pixel. We build on the concept of centered pyramids [4] and construct a set of locally centered image pyramids that span the scale-space of the image (Fig. 4, left).

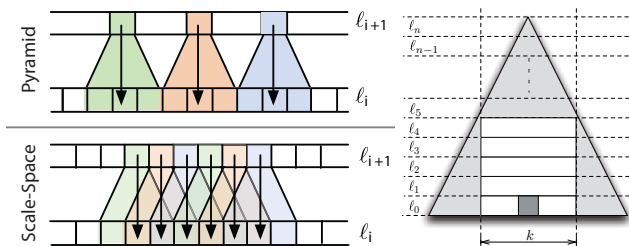


Figure 4. **Scale-Space** - *Left*: Image pyramids have an ambiguity in the parent-child relationship which is resolved by a complete scale-space. *Right*: For efficiency, we store only slices of the scale-space which are computed on-demand.

This scale-space representation is a set of Gaussian pyramids where each one is centered on a different pixel of the image. This gives a direct mapping from each pixel at  $\ell_0$  to a unique pixel at  $\ell_i$  for all levels  $i$ . We propose an efficient representation of this space. First, since we are only ever interested in a region-of-interest around each pixel (defined by the matching window), we do not need to compute or store the full image at each level, only a  $k \times k$  centered window as illustrated in Fig. 4 (right). Secondly, it may not be necessary to pre-compute each level for every pyramid, and so we propose to compute and store levels only on-demand. Using these slices of locally centered pyramids, we will next describe our signal equalization algorithm for correspondence matching.

### 4.3. Signal Equalization Algorithm

Our signal equalization algorithm takes as input two (or more) images of the same scene that can contain different spatial frequency signals of the same content, and uses local frequency maps and locally centered image pyramids to equalize the spatial frequencies prior to performing correspondence matching. The main idea of the algorithm is illustrated in Fig. 5.

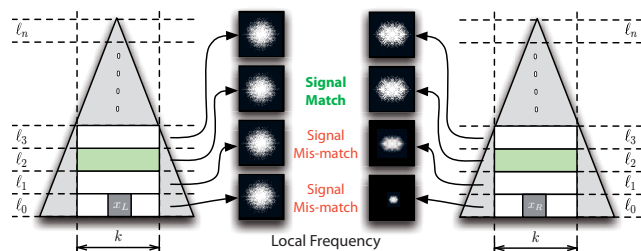


Figure 5. **Equalization Algorithm** - We continually traverse up the scale-space of two images centered on  $x_L$  and  $x_R$ , respectively, until the local frequency signals are equalized and optimal for correspondence matching.

We will describe the algorithm for the application of stereo correspondence of rectified images. Given two scanlines to match, the process starts by initializing the bottom level of the scale-space for each pixel on both scanlines.

Then for each potential correspondence  $(x_L, x_R)$  we determine if the highest local frequencies  $\tilde{\omega}_L$  and  $\tilde{\omega}_R$  are sufficiently high for matching. If either local patch does not meet the required frequency content, then we traverse up *both* respective pyramids one level and repeat the process until both patches contain sufficiently high frequencies. At this level ( $\ell_2$  in Fig. 5), the patches can be used directly for correspondence computation since they are locally centered on  $x_L$  and  $x_R$  respectively, they contain the optimal frequency content for matching, and they contain consistent frequency information. Note that since the frequency information is computed only locally, we use an iterative technique to proceed through the scale-space sequentially in practice, rather than immediately computing the final equalized signal as proposed in Eqn. 4.

## 5. Results

In this section we first evaluate the method on synthetic data to provide both qualitative and quantitative assessments of the performance. Second we provide two real-world examples to demonstrate the applicability of the approach to practical problems. All of our results are computed with a scale-space downsampling factor of 0.5.

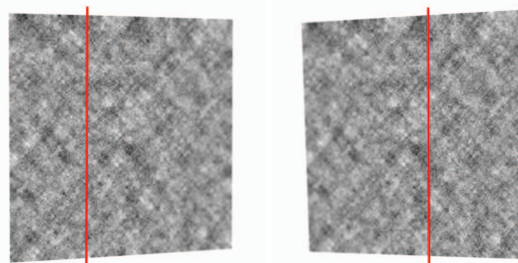


Figure 6. **Synthetic Plane** - Left and right images used for the synthetic experiment. The red lines indicate the approximate location of the focal plane of the cameras.

### 5.1. Synthetic Plane

To quantitatively evaluate the effect of the proposed signal equalization we generated a synthetic plane with wavelet noise texture and rendered it using the Mitsuba physically-based renderer [11] with simulated shallow depth of field. Fig. 6 shows the rendered images used as input to the algorithms with the approximate focal plane intersection shown in red.

We investigate three different algorithms. The first algorithm is a naïve implementation of window matching using normalized cross correlation. The second algorithm extends the naïve approach using the proposed signal equalization

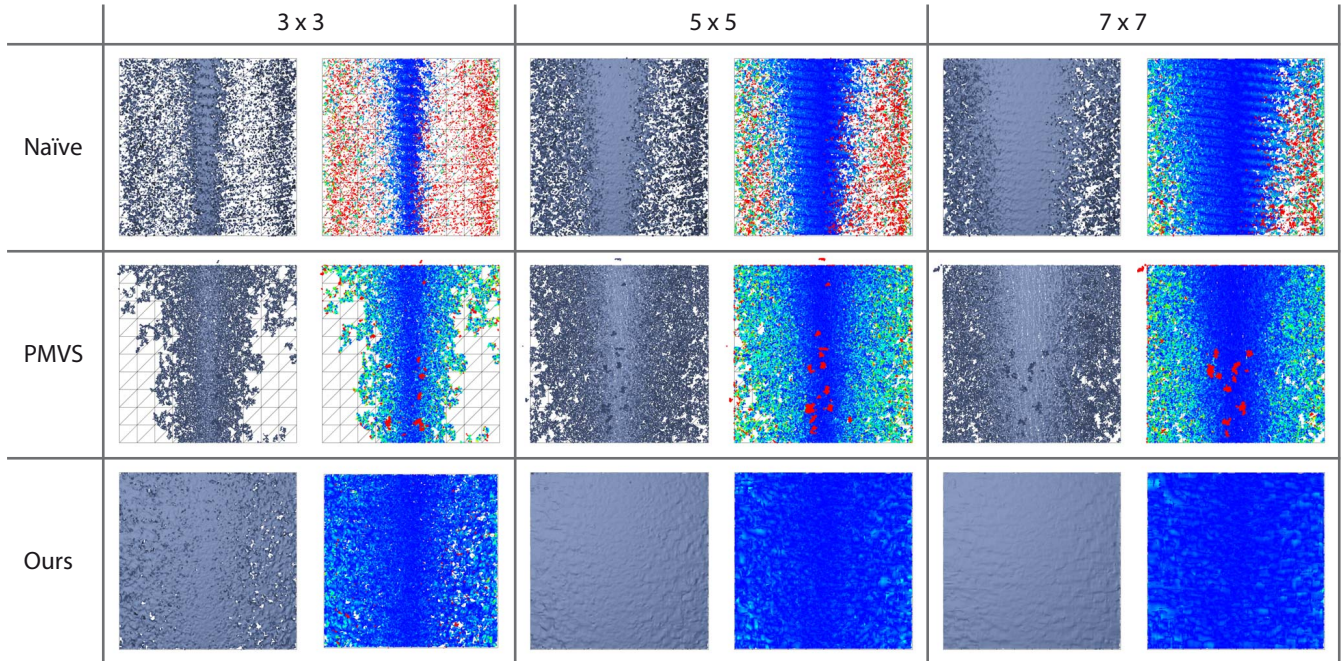


Figure 7. **Qualitative Evaluation** - Rendered point cloud results and error maps for the evaluated algorithms at three different window sizes. The error maps encode Euclidean distance to the plane in a rainbow scale, running from blue (low error) to red (high error).

	3x3				5x5				7x7			
	$\mu$	$\sigma$	#	%	$\mu$	$\sigma$	#	%	$\mu$	$\sigma$	#	%
Naïve	0.0675	0.1128	72,317	51	0.0376	0.0873	145,594	80	0.0160	0.0559	196,305	93
PMVS	0.0041	0.0244	21,868	49	0.0026	0.0036	39,061	83	0.0020	0.0026	221,688	89
Ours	0.0010	0.0011	220,969	96	0.0006	0.0005	236,282	98	0.0005	0.0004	236,306	98

Table 1. **Quantitative Evaluation** - Quantitative results for the synthetic example from Fig. 7. We give the mean absolute Euclidean distance ( $\mu \pm \sigma$ ) as a measure for the accuracy and the amount of points (#) as well as the percentage of area reconstructed (%) as a measure for the coverage of the algorithms.

framework. As a third algorithm we employ PMVS [7], a state-of-the-art multi-view stereo matching algorithm.

To evaluate the impact of the window size  $k$ , we run the experiments at three different window sizes ( $k = 3, 5, 7$ ). Increasing the window size increases the regularization of patch matching algorithms and thus improves their robustness as can be seen in Table 1, but at the same time reduces their spatial resolution [15].

Fig. 7 shows a qualitative evaluation and Table 1 lists quantitative measurements. To prune gross outliers, we computed the statistics in Table 1 on the 99% quantile - meaning we rejected the worst 1% of the points. The accuracy is measured as Euclidean distance to the plane. To compute the render and error maps shown in Fig. 7 as well as the coverage listed in Table 1 we splat the point clouds into the cameras using splat diameters of 3 pixels for PMVS and 2 pixels for the two other algorithms to account for the fact that PMVS produces sparser point clouds.

As expected, the naïve approach successfully reconstructs the surface where the two images have similar sig-

nals but fails when they differ. This is most apparent for  $k = 3$  but still holds with increasing window size. Signal equalization helps the naïve approach to overcome this limitation. Even for small window sizes, with signal equalization the algorithm successfully reconstructs almost the complete plane (96%) while increasing the accuracy by more than an order of magnitude. In fact, the improvement is so substantial that the naïve patch matching approach now even outperforms PMVS by a factor of  $4\times$  on degraded data. On non-degraded data all three methods have very similar accuracy, with PMVS being slightly better than the others. This indicates that signal equalization is likely to also improve the performance of sophisticated state-of-the-art matching algorithms.

Fig. 8 visualizes the level of the scale-space at which the matching occurred for each pixel. Darker areas correspond to matching at lower levels. With smaller window sizes matching is spread out over many layers in the scale-space because the frequencies present in such a limited neighborhood vary substantially across patches. Within larger win-

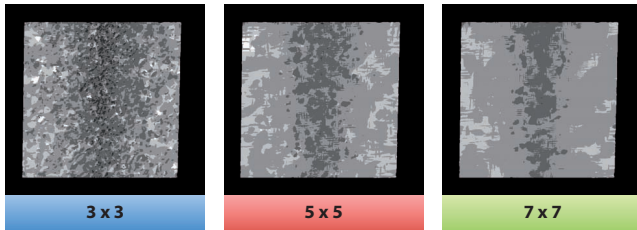


Figure 8. **Match Levels** - Visualizing the matching level for each pixel at three different window sizes. Darker areas correspond to matching at lower levels. For smaller window sizes the frequency content varies more between patches, leading to a wider spread in matching levels.

For smaller window sizes the variation of frequency content is lower and thus the matching levels are spatially more correlated, revealing the expected behavior that the matching level is a function of signal dissimilarity due to degradation.

## 5.2. Real-world Examples

As a first practical example, we demonstrate the improvement of signal equalization for the reconstruction of a human face from stereo views. Capturing living subjects typically poses a challenge since first the shutter speed should be high to avoid motion blur, second the aperture should be small to allow a large depth of field and third the amount of light that can be used before distracting the person is limited, especially when capturing performances. The result is typically a compromise and the captured imagery contains a limited depth of field. Fig. 9 (left) shows the two images acquired by the stereo rig as well as close-ups of various corresponding areas with different amounts of defocus blur. The center part of Fig. 9 visualizes the difference in local frequency content which is directly related to camera focus and shows the problem, where for camera  $C_2$  the neck is in focus (purple) while camera  $C_1$  focuses more on the cheek (yellow). The area colored in cyan has similar degradation in both images and corresponds to the area the naïve approach succeeds at reconstructing. Signal equalization greatly extends the reconstructable area and improves reconstruction quality as depicted in Fig. 9 (right).

A second example is stereo reconstruction from macro photography. If we wish to capture small objects in high resolution it is common to use a macro lens, which is specifically designed for close-up shots but suffers from incredibly small depth of field. Inevitably, a stereo rig of macro images will contain different defocus regions. An example is shown in Fig. 10 (left), including close-ups of several corresponding areas with different amounts of defocus blur. Again, our signal equalization approach greatly improves reconstruction quality as shown in Fig. 10 (right).

## 6. Conclusion

In this paper we aim to alleviate correspondence errors when matching two images that differ in signal content due to degradation. We focus on patch-based matching using normalized cross-correlation, since this is one of the most widely used metrics in computer vision. Our approach is to locally equalize and optimize image signals before computing correspondence matches. This is made possible by a local frequency estimation technique and an efficient scale-space image representation.

In our current implementation, we model signal degradation as isotropic lowpass filtering. However, for situations such as motion blur a more suitable degradation function should be used. We consider this to be future work. Another interesting venue for future research would be to use the differently degraded signals after matching to recover the original signal as well as possible. This could allow to compute all-focus texture maps from partially defocused cameras [17], or to enhance the geometry using shape-from-shading cues [2].

We have described and evaluated our technique in the context of improving dense stereo reconstruction given images that are degraded by different amounts of defocus blur due to low depth of field. However, correspondence matching is a fundamental tool in computer vision, and our algorithm can be applied in a variety of other contexts. An example would be to equalize frame-to-frame signals for optical flow estimation or video stabilization that have been degraded by time varying blur, such as motion blur or focus change. Another example is to equalize local signals for template matching or image registration. Furthermore, we have formulated our approach as a preprocess that modifies the input data and can thus be used in combination with a wide variety of existing powerful correspondence techniques. As a result, we believe this work has the potential to impact a broad range of applications in computer vision.

## References

- [1] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 92:1–31, 2011.
- [2] T. Beeler, B. Bickel, R. Sumner, P. Beardsley, and M. Gross. High-quality single-shot capture of facial geometry. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 2010.
- [3] D. Bradley, T. Boubekeur, and W. Heidrich. Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In *CVPR*, 2008.
- [4] P. Brigger, F. Müller, K. Illgner, and M. Unser. Centered pyramids. *IEEE Trans. Image Proc.*, 8(9), 1999.
- [5] B. Cyganek. Adaptive window growing technique for efficient image matching. *LNCS*, 3522:308–315, 2005.
- [6] J. Ens and P. Lawrence. An investigation of methods for determining depth from focus. *IEEE PAMI*, 15:97–108, 1993.

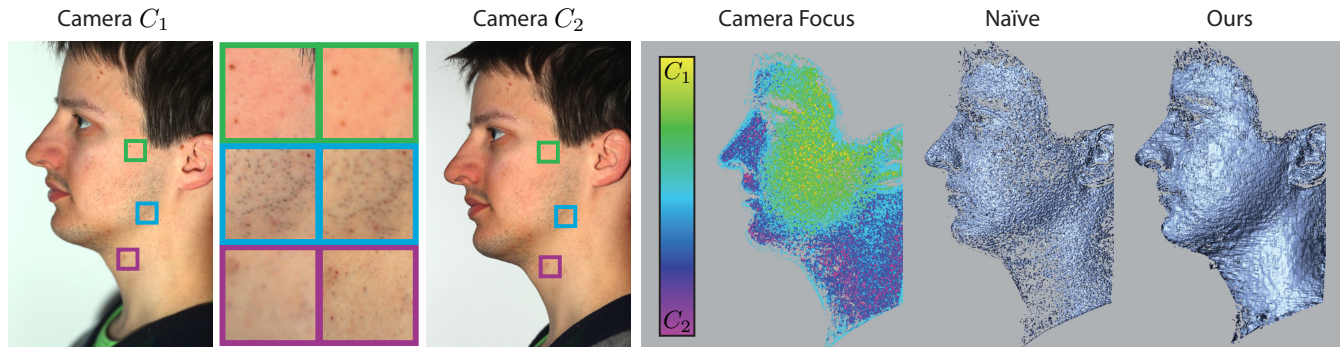


Figure 9. **Real-world Face Example** - *Left*: The images from cameras  $C_1$  and  $C_2$  have different focal planes with limited depth of field. *Center*: Color-coded difference of local frequencies. *Right*: 3D point cloud results. The naïve algorithm successfully reconstructs areas where the signals are similar but quickly degrades with increasing signal mismatch. Signal equalization greatly improves accuracy and robustness in areas of signal mismatch.



Figure 10. **Real-world Macro Example** - *Left*: The captured images have different focal planes with extremely limited depth of field. *Center*: Color-coded difference of local frequencies. *Right*: 3D point cloud results from two different views.

- [7] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *IEEE Trans. PAMI*, 32(8):1362–1376, 2010.
- [8] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. In *ICCV*, 2007.
- [9] M. Habbecke and L. Kobbelt. A surface-growing approach to multi-view stereo reconstruction. In *CVPR*, 2007.
- [10] X. Hu and P. Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE PAMI*, 34(11), 2012.
- [11] W. Jakob. Mitsuba, 2012. [www.mitsuba-renderer.org](http://www.mitsuba-renderer.org).
- [12] H. Jin and P. Favaro. A variational approach to shape from defocus. In *ECCV*, 2002.
- [13] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Trans. PAMI*, 16(9), 1994.
- [14] R. Klowsky, A. Kuijper, and M. Goesele. Modulation transfer function of patch-based stereo systems. In *CVPR*, 2012.
- [15] R. Klowsky, A. Kuijper, and M. Goesele. Weighted patch-based reconstruction: linking (multi-view) stereo to scale space. *LNCS*, 7893, 2013.
- [16] H.-S. Koo and C.-S. Jeong. An area-based stereo matching using adaptive search range and window size. *LNCS*, 2074:44–53, 2001.
- [17] S. Li, J. T. Kwok, and Y. Wang. Combination of images with diverse focuses using the spatial frequency. *Information Fusion*, 2(3):169 – 176, 2001.
- [18] C. Menard and W. Kropatsch. Adaptive stereo matching in correlation scale-space. *LNCS*, 1310:677–684, 1997.
- [19] A. S. Ogale and Y. Aloimonos. Shape and the stereo correspondence problem. *IJCV*, 65(3):147–162, 2005.
- [20] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1/2/3):7–42, 2002.
- [21] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006.
- [22] C. Strecha, A. M. Bronstein, M. M. Bronstein, and P. Fua. Ldhash: improved matching with smaller descriptors. *IEEE Trans. PAMI*, 34(1):66–78, 2012.
- [23] M. Subbarao and G. Surya. Depth from defocus: A spatial domain approach. *IJCV*, 13(3):271–294, 1994.
- [24] S. Yoon, D. Min, and K. Sohn. Fast dense stereo matching using adaptive window in hierarchical framework. *LNCS*, 4292:316–325, 2006.