A Toolbox to Visualize Dense Image Correspondences (Stereo Disparities & Optical Flow)

A. Sellent, P.-S. Lauer, D. Kondermann, J. Wingbermühle

February 7, 2012

1 Motivation

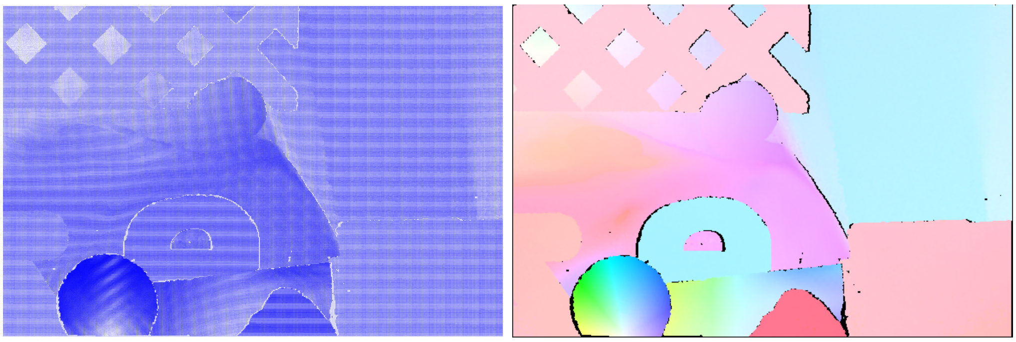
Today many different algorithms to estimate optical flow or stereo correspondences between images are published. This makes visualization of the results in a comparable fashion an important issue. The first evaluation of an algorithm is always visually, to ensure that the estimated correspondence field approximates the intuitively expected pixel displacement. As a second step, usually some comparison to known ground-truth correspondences is performed where also quantitative evaluation is possible, e.g. for the data-sets published by Baker et al. [5]. However, the set of scenes with known ground-truth correspondences is very restricted, either to very simple scenes or to contrived or synthetic scenes with limited realism. In contrast, human observers have a good understanding of the displacement in arbitrary real world image pairs. If a standardized visualization of correspondence fields is available, the intuitive motion estimation ability of humans can be exploited.

Current visualization of correspondences often use amplitude-dependent scalings which depend on the maximal flow or the maximal disparity that has been estimated for one image pair [3]. So the visualization might show outliers very clearly, but single outliers might render the remaining correspondences quite indistinguishable. Even more importantly, the visual comparison of the output of different algorithms becomes highly dependent on the presence and the value of outliers. We therefore implemented some different visualization schemes with common thresholds that allow for outlier-visualization on the one hand and fine-scale correspondences visualization on the other hand. These schemes can also be applied to real world correspondences and used for the visual comparison of different algorithms.

2 Desired Properties

To assess the quality of correspondences by their visualization, several properties are desirable.

1. Visualization should be capable to show the results for every pixel.



(a) (b)

Figure 1: Visualization of the flow field *Rubber Whale* from the Middlebury benchmark [5] using (a) an arrow-plot or (b) a scaled color map [4].

2. Even single pixel outliers should be visible.
3. Small errors within regions of constant correspondences should be visible.
4. For presentation within limited space only a small number of images should be required to fully assess the quality of a correspondence field.
5. The visualization should allow to judge how accurately discontinuities in the correspondence estimation and object-boundaries coincide.

3 State-of-the-Art

In the literature, several methods for correspondence visualization are common.

Arrow Plots To each pixel an arrow is assigned that corresponds to the displacement of this pixel, Fig. 1a. This renders the direction and extend of the correspondence immediatly accessable. However, in large images or images with large displacements (greater than one or two pixels) arrow-plots quickly grow confusing. A common remedy is to subsample the arrows [6]. However, this rejects information for many pixels and single-pixel outliers can be easily missed.

Gray-Value Coding A very common way to visualize stereo disparities is via gray-value coding using light values for close objects and dark gray-levels for distant objects. In this way, at most 255 different disparity levels can be displayed.

Gray value coding for flow fields is less common, as flow fields have a horizontal and a vertical component to be displayed. Sometimes, only the magnitude of the optical flow is encoded into the gray-value and the information on the flow direction is suppressed.

Color-Coding Schemes In stereo correspondence analysis color coding schemes allow for the dense visualization of more than 255 levels. The increase of levels is suited much better for sub-pixel accurate algorithms than the gray-value coding.

In optical flow analysis color-coding schemes allow to visualize horizontal and vertical component in one image. Colors can be assigned to each pixel without interference due to large displacements. However, most color coding schemes use mapping from correspondences to color that depend on the minimal and maximal value in the correspondence field, [5] While this representation is highly suitable to exploit the full color-space for visualization, it renders comparison of different flow algorithms extremely difficult. The presence and amount of very few outliers is able to influence the visualization considerably.

Color visualization has the drawback that it is expensive to reproduce in hard-copy print. But as scientific results are more and more accessed via color monitors, color-coding schemes seem a suitable way to visualize correspondences.

4 Visualization Concepts

In the implemented visualization, we use the basic ideas of color coding, as it allows to represent sub-pixel correspondence in 1 or 2 dimensions concisely. To be robust towards outliers, we propose to scale the color-code independently of the input. This fixed mapping from correspondences to color allows to provide a legend that assigns a fixed correspondence length and direction to a certain color, Fig. 2. For optical flow estimation, the legend gives the human observer an idea of the direction as well as the length of the estimated flow. However, this method has a drawback: Outliers may extend beyond the color-range and become indistinguishable from correctly estimated correspondences. In the other extreme, small variations, e.g., within an object, might become indiscernible due to the fixed scale of the color-map.

To increase the range of visualized correspondences, we propose a second encoding. This additional encoding iterates cyclically over all colors. Providing a sufficiently short cycle-length also small variations of the correspondences can be visualized. Although large scale outliers might have a cyclic representation that coincides with the surrounding correspondences, the probability for this is rather small if the full color-spectrum is exploited for the cyclic representation. To obtain a comparable visualization, the cycle-length should be maintained over all visualizations.

The two above presentations are complementary in focusing on the general correspondence impression with the color-coding and a fine-scale or outlier impression with the cyclic-coding. We also offer a third visualization that is to supplement the fixed-scale color-coding. In this visualization the correspondences are adjusted by given values before the color-coding scheme from above is employed. The adjustment values might, e.g. , be the component-wise mean of an optical flow field or the minimum and maximum value of a disparity field. In the case of mean adjustment, optical flow fields are centered around zero motion. In the case of given minimum and maximum disparity value, the color map can be scaled to fit the disparity field. In both cases the color map can be

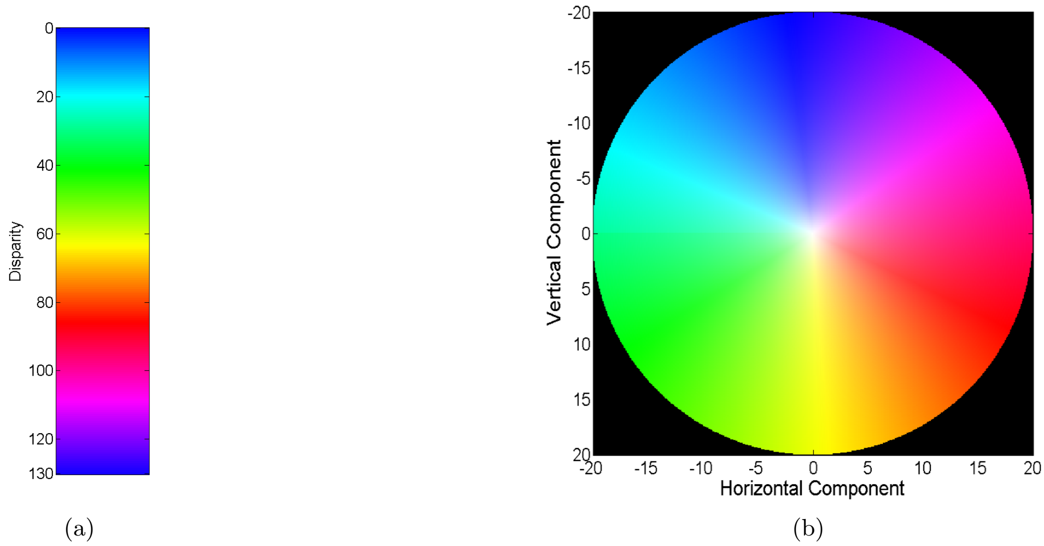


Figure 2: Standardized visualization is equipped with a legend that assigns each color to (a) a specific disparity or (b) a specific optical flow vector.

better exploited. While this is very helpful when large global offsets dominate the correspondence field, this visualization is only comparable if all analyzed correspondence fields are adjusted by the same values. Therefore the spreading-parameters should be clearly indicated in connection to the visualization image.

All of the aforementioned visualizations are fine-grained visualizations that can cope with the desired properties 1- 4. They can also indicate discontinuities in the correspondence field by sharp color-discontinuities. However, separate visualization of correspondence field and underlying image is not able to indicate the accuracy of the location of discontinuities. Best assessment of this property can be achieved by superimposing correspondence visualization and images. Note however that the superposition might conceal information on the correspondence fields, Sect. 7.

5 Implementation Details for Stereo Visualization

For the visualization of stereo-image correspondences we assume the image pair to be rectified, else the 2D representation of the optical flow field is used. In the rectified case, the disparity can be given as one matrix on the image domain $\Omega \subset \mathbb{N} \times \mathbb{N}$. Although stereo correspondences are often estimated at full-pixel accuracy, we assume the disparity function $d : \Omega \to \mathbb{R}$ to take arbitrary positive values. In all visualizations we additionally allow for the label *unknown* and represent these pixels as black.

5.1 Fixed Color-Coding

As a first step of the color-coding, we clip the disparity at the maximal value of 130 pixels. This threshold was chosen to be well adjusted to the range of a stereo setup such as described by the HCI-benchmark on [1], however, different camera-setups and different image sizes might require to reset the threshold. This resetting should be indicated clearly, as it influences the entire mapping of disparity values to color as given in Fig. 2.

We then compress the range of the disparity values: The estimation of small disparities needs to be more accurate than the estimation of large disparities to obtain constant accuracy of the depth estimation, as depth is inverse-proportional to the disparity. For ease of implementation we use the compression function

$$\phi_\gamma : \mathbb{R} \to \mathbb{R}; x \mapsto sign(x)\ |x|^\gamma$$

with fixed $\gamma = 0.95$.

After compression, the disparity is mapped to the color-map. We reorder the hue component of the HSV-color space [8] so that reddish colors indicate close-by objects and dark blue indicates objects at infinity. If the disparity does not correspond exactly to one color of the map, we use linear interpolation.

5.2 Cyclic Representation

For the cyclic representation we repeat the same color-map each 20 pixels of the uncompressed disparity.

5.3 Full Color-Map Representation

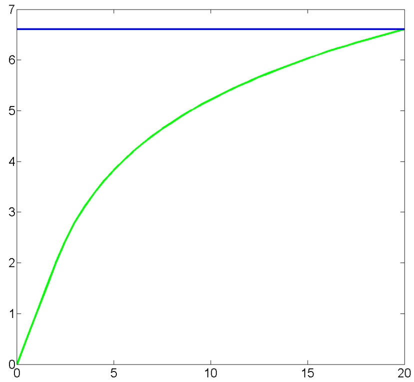
For the third visualization, a minimal and maximal disparity is provided by the user. We subtract the minimal disparity value from the estimated disparity and compress the difference with $\gamma = 0.95$. Values are not clipped to a valid range and if estimated disparity values extend over the maximal or minimal given disparity, the color-spectrum is repeated cyclically.

6 Implementation Details for Optical Flow Visualization

Flow fields are usually provided as two matrices on the pixel grid $\Omega \subset \mathbb{N} \times \mathbb{N}$, one for each component of the flow field. Flow components are generally assumed to be mappings $u : \Omega \to \mathbb{R}$ and $v : \Omega \to \mathbb{R}$ which can take arbitrary real values. In all visualizations we additionally allow for the label *unknown* and represent these pixels as black.

6.1 Fixed Color-Coding

As in the stereo coding, we first clip the range of the flow values. We set the clipping threshold to 20 pixels motion magnitude, as in scenes such as provided by the HCI-sequences [1] the magnitude of the flow is often below this value. Different image sizes



(a)

Figure 3: Logarithmic scaling leaves values smaller than $T = 2$ untouched and approaches the full-saturation line (blue) less abruptly than linear encoding.

and frame-rates might require to reset the threshold. However, this should be indicated clearly.

Usually, optical flow is estimated and evaluated at sub-pixel accuracy. Sub-pixel values are often based on interpolation, and are insignificant for some applications. We therefore use the function

$$\psi(x) = \begin{cases} x & \text{for } |x| < T \\ sign(x)T(1 + \log(\frac{|x|}{T})) & \text{else} \end{cases}$$

with $T = 2$ as a compression function. In contrast to the compression function ϕ_γ used for stereo, ψ leaves small values untouched and compresses large values smoothly, Fig. 3a.

The color of the pixel is determined by the angle of its flow vector with the horizontal image axis. We map angle equidistantly to a shifted HSV color-map. The color shift is performed so that purely downward motion is encoded in yellow which yields a high contrast to black asphalt in common images of driver assistant scenarios. If the angle does not correspond exactly to one color of the map, we use linear interpolation.

The second dimension of the motion - its extent - is encoded by the saturation of the color. Small magnitude motion is encoded in white and pastel colors and saturation increases with the magnitude of the motion, until the full saturation is reached at the threshold of 20 pixels.

6.2 Cyclic Representation

For the cyclic representation we only consider the magnitude of the motion. We repeat the same color-map as above with a cycle length of 10 pixel of motion magnitude. This representation encodes only the magnitude and does not enable to show small variations of the direction. However, the color-saturation plot visualizes directional deviation quite clearly in color, while the variation in magnitude with its representation in saturation is harder to assess. We therefore consider the cyclic representation a suitable complement to the other visualization methods.

6.3 Full Color-Map Representation

To better exploit the color-map, we also provide a component-wise adjusted visualization. As default, the component-wise means of the motion can be used for adjustment. However, for better comparability between algorithms, the visualization also accepts user-provided values. The adjusted flow fields are then processed as in Sect. 6.1.

7 Overlay of Visualizations and Images

The above visualization cannot suffice the desired property 5 to judge the alignment of correspondence discontinuities with object boundaries in the image. We therefore provide an overlay of correspondence visualization and the original image. While gray-value images and stereo disparities can be easily combined, the combination of optical flow visualization and the original images might obscure some information on the flow field. In the flow visualization of Sects. 6.1 and 6.3, the magnitude of the flow is encoded into the saturation of the color, so variation in the overlay image may either be due to gray-value variation in the image or magnitude variation in the flow field.

We therefore recommend to use overlay images only with stereo correspondence visualization and the cyclic optical flow visualization, as here saturation is not relevant for correspondence encoding.

8 Visualization Examples

As examples for our visualization tools, we visualize the correspondences estimated on one of the less challenging image pairs from the set provided at [**?**], i.e. the first stereo and the first optical flow pair of the *Sunflare* sequence, before the sun enters into the camera. We use the stereo algorithm of Hirschmüller [7] and the optical flow algorithm of Sun et al. [9] with default parameters to estimate correspondences for these images. The stereo visualization and overlay images in Fig. 4 show the disparity equally well, while the latter also admit for evaluation of the location of discontinuities. The depth structure and the depth of distant objects is more clearly visible at the scaled images in 4c, however, this representation is less suitable for inter-algorithm comparison. Fig. 4 also shows that the cyclic representation is unsuitable to convey a good impression on the global quality of the disparity but that small variations in disparity are emphasized.

The global flow representation in Fig. 5a gives a good overall impression of the global sideway motion in the scene but fails to distinguish between objects that move with only slightly different velocities. More details are revealed in the cyclic representation

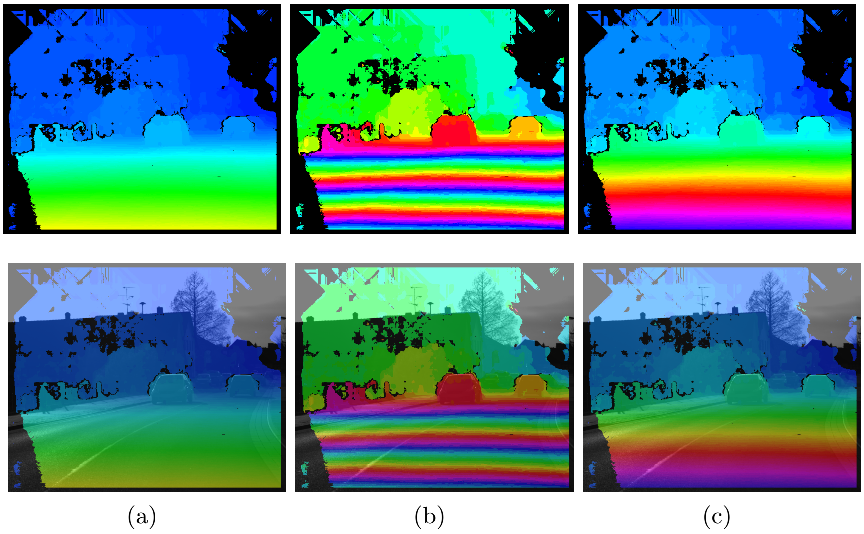


Figure 4: Stereo correspondence visualization (top row) and the same visualization as overlay over the original reference image (bottom row). (a) The fixed scaling allows for comparable visualization, but does not exploit the full color-space. (b) Cyclic encoding highlights small variations of the correspondences while (c) stretching between the extreme values 1 and 64 allows to exploit the color-space more thoroughly but looses easy interpretation with the legend in Fig. 2.

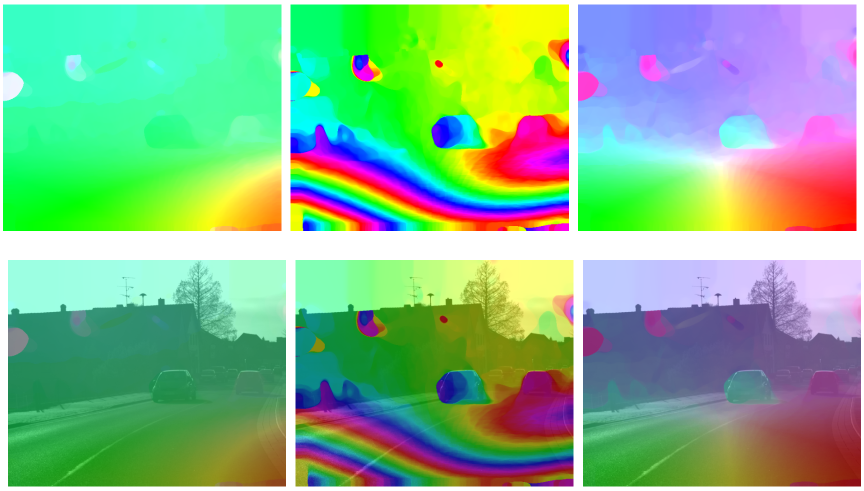
which also highlights outliers in some of the image regions clearly. While motion is best represented in the mean-adjusted variant 5b, this representation does not allow for inter-algorithm comparison.

In Fig. 5 we also remark that overlay with the original images tend to be confusing for the assessment of the two components of optical flow. While the overlay representation is essential for the assessment of the discontinuities the visualizations in the top row give a much better representation of the motion.

9 Conclusion

We here motivate the choices we considered in designing our tools for the visualization of dense image correspondences. The visualizations have been realized in MATLAB and C++ (thanks to Anita Sellent and Paul-Sebastian Lauer for the implementation) and can be downloaded under a public license from [2].

Suggestions for improvements are welcome, especially if they concern complementary ideas to highlight as yet unconsidered properties of correspondence fields.



(a) (b) (c)

Figure 5: Optical flow visualization (top row) and the same visualization as overlay over the original reference image (bottom row). (a) The fixed scaling allows for comparable visualization, but does not exploit the full color-space. (b) Cyclic encoding highlights small variations of the correspondences while (c) mean-adjusting the horizontal field by -8.60 pixels and the vertical field by 2.96 pixels allows to exploit the color-space more thoroughly but looses easy interpretation with the legend in Fig. 2.

References

[1] http://hci.iwr.uni-heidelberg.de//benchmarks/document/challenging_data_for_stereo_and_optical_flow/.

[2] http://hci.iwr.uni-heidelberg.de//benchmarks/media/correspondencevisualization/index.html.

[3] http://vision.middlebury.edu/.

[4] http://vision.middlebury.edu/flow/.

[5] S. Baker, D. Scharstein, JP Lewis, S. Roth, M.J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011.

[6] J.L. Barron, D.J. Fleet, and SS Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[7] H. Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 807–814. IEEE Computer Society, 2005.

[8] A.R. Smith. Color gamut transform pairs. In *SIGGRAPH Computer Graphics*, volume 12, pages 12–19. ACM, 1978.

[9] D. Sun, S. Roth, and M.J. Black. Secrets of optical flow estimation and their principles. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2432–2439. IEEE, 2010.