Tutorial

Bidirectional Texture Functions Modelling

Analysis & Compression

Michal Haindl Jiří Filip

Institute of Information Theory and Automation
of the AS CR

13/06/2010
CVPR 2010

www.utia.cz

Outline of Analysis and Compression

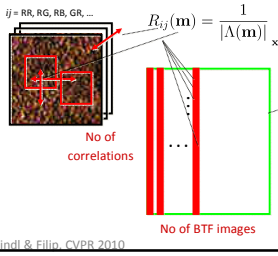
- BTF data dimensionality analysis
  - SVD, PCA & psychophysical methods
- Methods of BTF data compression
  - PCA Eigen-images
  - Reflectance models
  - SVD tensors
  - Spherical harmonics + wavelets
  - Data clustering
  - Layered volumetric models
  - Vector quantization
- Selected methods comparison

BTF Sample Dimensionality Analysis

[Suen & Healey, IEEE PAMI 2000]

SVD of BTF data correlations

- Sample complexity ~ Dimensionality ⇨ Number of highest eigen-values of correlation between RGB components in moving support window (for each combination of view and illumination directions).
- Tested on CURET database only.



$$R_{ij}(\mathbf{m}) = \frac{1}{|\Lambda(\mathbf{m})|} \sum_{\mathbf{x}\in\Lambda,(\mathbf{x}+\mathbf{m})\in\Lambda} [I_i(\mathbf{x}) - \bar{I}_i][I_j(\mathbf{x}+\mathbf{m}) - \bar{I}_j]$$

$$\bar{I}_i = \frac{1}{|\Lambda|}\sum_{\mathbf{x}\in\Lambda} I_i(\mathbf{x})$$

$$\mathbf{x} = [r_1, r_2]^T$$

$$\mathbf{R} = \mathbf{UDV}^T$$

$$\frac{\sum_{i=1}^{\mathcal{D}} \sigma_i^2}{\sum_{i=1}^{n_i n_v} \sigma_i^2} \geq F$$

Typical values 5-15 ($F = 0.99$)

BTF Sample Dimensionality Analysis

[Koudelka et al. TEXTURE 2003]

PCA of BTF data

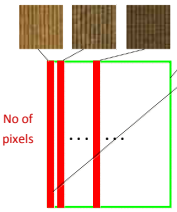
- Dimensionality ⇨ number of eigen-images with the highest energy
- Typical value 30 ($F = 0.9$)

$$\mathbf{A} = [\mathbf{X}_1 - \bar{\mathbf{X}}, \mathbf{X}_2 - \bar{\mathbf{X}}, \ldots, \mathbf{X}_{n_i \cdot n_v} - \bar{\mathbf{X}}]$$

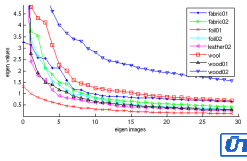
$$\mathbf{M} = \mathbf{A}^T\mathbf{A}$$

$$\mathbf{M} = \mathbf{UDV}^T$$

$$\frac{\sum_{i=1}^{\mathcal{D}} \sigma_i^2}{\sum_{i=1}^{n_i n_v} \sigma_i^2} \geq F$$



Mean BTF image



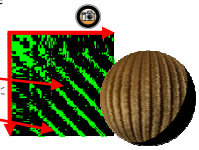
BTF Sample Dimensionality Analysis

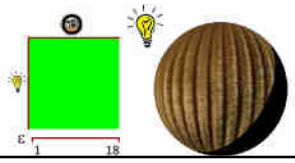
[Filip et al. ACM TOG 2008]

Psychophysical estimation of BTF dimensionality

- Pixel-wise difference of image pairs (SSIM, VDP, CIE LAB, etc.)
- Similar images substituted by representative one
- Selection of important BTF images only
- **BTF index map**:
  - ■ Substituted images (index only)
  - ■ Preserved BTF images

#■ >>#■



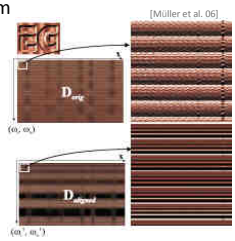


Selected subset:
5-30% of original images

BTF Data Alignment Dimensionality Reduction

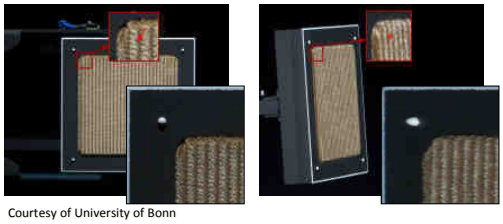
[Müller et al. 06] – BTF variance reduction by data alignment with respect to a local geometry.

- Estimate per-pixel coordinate system maximizing ABRDFs correlation ⇨ rotations of ABRDFs.
- Maximization ⇨ ABRDFs represented by spherical harmonics in Fourier space.
- Variance reduction ⇨ higher compression



Compression Methods

- Compression of all BTF images at once ($BTF(r_1, r_2, i, v)$)



Courtesy of University of Bonn

- Compress individual views (reflectance fields $\mathcal{R}_v(r_1, r_2, i)$) separately

Compression Methods Taxonomy

- Compression of all BTF images at once
  - ABRDF factorization
  - Fitting BRDF models to pixel-wise ABRDFs
  - Full BTF factorization
  - Per-cluster factorization
  - ABRDFs clustering + clusters compression (SVD,BRDF model)
- Compression of separate views (reflectance fields)
  - Fitting BRDF models to a fixed view ABRDF data
  - Per-view factorization
- Combining geometrical and reflectance information
- Multi-level vector quantization

ABRDF Factorization Approaches

[Kautz et al. 99] – use ***SVD*** to produce two 2D factors instead of 4D BRDF.

$$BTF(r_1, r_2, i, v) \approx \sum_{k=1}^{K_j} P_{k,r_1,r_2}(\pi_1(\omega_i, \omega_v)) Q_{k,r_1,r_2}(\pi_2(\omega_i, \omega_v))$$

[McCool et al. 01] – use ***Homomorphic factorization*** to generate more than two positive factors.

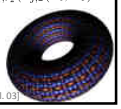
$$BTF(r_1, r_2, i, v) \approx \prod_{j=1}^{J} P_{j,r_1,r_2}(\pi_j(\omega_i, \omega_v))$$

[Suykens et al. 03] – each pixel = product of three or more two-dimensional positive factors using ***chained matrix factorisation***.

$$BTF(r_1, r_2, i, v) \approx \prod_{j=1}^{J} \sum_{k=1}^{K_j} P_{j,k,r_1,r_2}(\pi_{j,1}(\omega_i, \omega_v)) Q_{j,k,r_1,r_2}(\pi_{j,2}(\omega_i, \omega_v))$$

☑ factors in form of textures ⇨ interactive rendering

☒ low compression $\approx \frac{1}{100}$



[Suykens et al. 03]

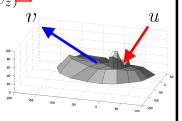
Fitting BRDF Reflectance Models

Fitting Lafortune lobes [Lafortune et al. 07] to BRDF.

- One lobe model

$$Y_{i,v} = \rho[\omega_i^T \mathbf{D} \omega_v]^n = \boxed{\rho}\,\boxed{D_x} u_x v_x + \boxed{D_y} u_y v_y + \boxed{D_z} u_z v_z)^{\boxed{n}}$$

  - 5 parameters instead of *many reflectance values*



- More reflectance lobes
  - Better fitting, more difficult parameters estimation

$$Y_{i,v} = \sum_{k=1}^{n_l} \rho_k (D_{x,k} u_x v_x + D_{y,k} u_y v_y + D_{z,k} u_z v_z)^{n_k}$$

- Properties
  - Generalization of cosine lobe model
  - Physically plausible
  - Non-linear iterative estimation of parameters, depends on initialization
  - Memory efficient representation of BRDF

Fitting BRDF Reflectance Models

[McAlister 02] – fitting pixel-wise single multi-lobe Lafortune model (LM) to all view and illum. directions

$$BTF(r_1, r_2, i, v) \approx \rho_{d,r_1,r_2} + \sum_{k}^{n_l} \rho_{s,r_1,r_2,k} [\omega_i^T \mathbf{D}_{r_1,r_2,k} \omega_v]^{n_{r_1,r_2,k}}$$

[McAllister 02]

☑ good compression, fast fitting

☒ blurry images, suitable for low-height samples only

[Daubert 01] – LM + additional iteratively obtained look-up table

$$BTF(r_1, r_2, i, v) \approx T_{\omega_v, r_1, r_2} \left(\rho_{d,r_1,r_2} + \sum_{k}^{n_l} [\omega_i^T \mathbf{D}_{r_1,r_2,k} \omega_v]^{n_{r_1,r_2,k}} \right)$$

☑ good quality for regular samples

☒ require to store extensive view-dependent look-up table

[Daubert 01]

Fitting Per-View BRDF Reflectance Models

[Meseth et al. 03]

[Meseth et al. 03] – Colour albedo-map modulated by a luminance Lafortune model (up to 3 lobes)

☑ less parameters to store

☒ for less specular samples only

$$\mathcal{R}_v(r_1, r_2, i, v) \approx \rho_{d,r_1,r_2} + \rho_{s,r_1,r_2,k} \sum_{k}^{n_l} [\omega_i^T \mathbf{D}_{r_1,r_2,k}]^{n_{r_1,r_2,k}}$$

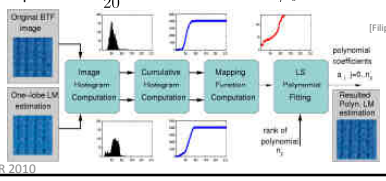
[Filip & Haindl 04] – polynomial extension of Lafortune model

☑ good visual quality

☒ lower compression $\approx \frac{1}{20}$

$$\mathcal{R}_v(r_1, r_2, i, v) \approx \sum_{j=0}^{n_p} a_{r_3,v,i,j} [\rho_v(r)(\omega_i^T \mathbf{D}_v(r))^{n_v(r)}]^j$$

[Filip & Haindl 04]



Fitting Per-View BRDF Reflectance Models

[Malzbender 01] – each view direction represented by per-pixel polynomials.

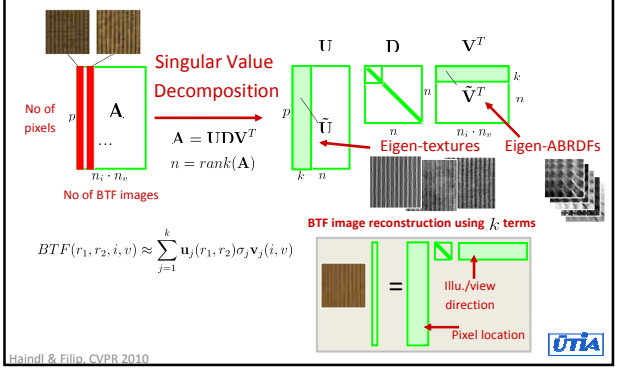
$$\mathcal{R}_v(r_1,r_2,i) \approx a_o(r_1,r_2)u_{r_1}^2 + a_1(r_1,r_2)u_{r_2}^2 + a_2(r_1,r_2)u_{r_1}u_{r_2} + a_3(r_1,r_2)u_{r_1} + a_4(r_1,r_2)u_{r_2} + a_5(r_1,r_2)$$

☑ fast fitting

☒ low quality, not suitable for specular materials

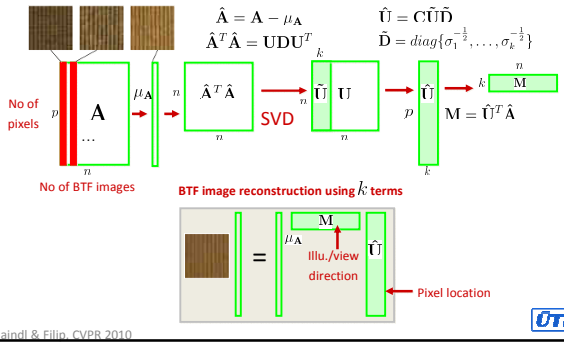
[Malzbender 01]

SVD-Based BTF Factorization



$$BTF(r_1,r_2,i,v) \approx \sum_{j=1}^{k} \mathbf{u}_j(r_1,r_2)\sigma_j\mathbf{v}_j(i,v)$$

PCA-Based BTF Factorization



$$\hat{\mathbf{A}} = \mathbf{A} - \mu_{\mathbf{A}} \qquad \hat{\mathbf{U}} = \mathbf{C}\tilde{\mathbf{U}}\tilde{\mathbf{D}}$$

$$\hat{\mathbf{A}}^T\hat{\mathbf{A}} = \mathbf{U}\mathbf{D}\mathbf{U}^T \qquad \tilde{\mathbf{D}} = diag\{\sigma_1^{-\frac{1}{2}},\ldots,\sigma_k^{-\frac{1}{2}}\}$$

$$\mathbf{M} = \hat{\mathbf{U}}^T\hat{\mathbf{A}}$$

Full BTF Factorization

[Koudelka et al.03]

[Koudelka et al. 03] – PCA of all BTF images in texture space.

$$BTF(r_1,r_2,i,v) \approx \sum_{k=1}^{n_c} \mathbf{u}_k(i,v)\sigma_k\mathbf{v}_k(r_1,r_2)$$

☑ high compression $\approx \frac{1}{1000}$, quality control

☒ long reconstruction times, lower quality

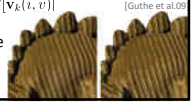
[Guthe et al. 09] – PCA of all BTF images in ABRDF space.

- Additional transformation $\boldsymbol{T}$ to minimize visual difference
- Visual difference ⇨ spatio-temporal filtering ⇨ HVS cone responses, contrast sensitivity function

$$BTF(r_1,r_2,i,v) \approx \sum_{k=1}^{n_c} T[\mathbf{u}_k(r_1,r_2)]\sigma_k T[\mathbf{v}_k(i,v)]$$

☑ high compression $\approx \frac{1}{500}$, perceptual quality measure

☒ long compression times



[Guthe et al.09]

Per-View BTF Factorization

[Sattler et al. 03] – PCA of all BTF images in individual view directions.

$$\mathcal{R}_v(r_1,r_2,i) \approx \sum_{k=1}^{n_c} \mathbf{u}_{v,k}(i)\sigma_k\mathbf{v}_{v,k}(r_1,r_2)$$

☑ fast compression & reconstruction, good visual quality

☒ low compression rates $\approx \frac{1}{20}$

[Sattler et al.03]

[Wong & Leung 03] – compress illumination dependent data ⇨ ***spherical harmonics***. The coefficient planes further coded using a discrete wavelet transformation.

☑ reasonable compression, controllable by a number of components

☒ lower reconstruction speed – requires inverse DWT

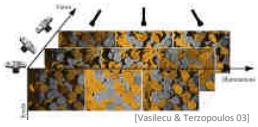
Full BTF Factorization

[Vasilescu & Terzopoulos 03] – BTF images as vectors in 3D tensor ⇨ multi-modal SVD ⇨ controllable compression of view/illum. effects.

$$\mathcal{B} = \mathcal{Z} \times_1 \mathbf{U}_{tex} \times_2 \mathbf{U}_i \times_3 \mathbf{U}_v$$

$$\mathcal{T} = \mathcal{B} \times_2 \mathbf{U}_i \times_3 \mathbf{U}_v$$

$$BTF(i,v) = \mathcal{T} \times_2 \mathbf{i}^T \times_3 \mathbf{v}^T$$



[Vasilecu & Terzopoulos 03]

[Wang et al. 04] – Instead of 3D texel-illumination-view tensor, BTF data directly in 4D tensor ⇨ preserving spatial relationships in individual BTF images.

☑ compact representation, controllable compression $\approx \frac{1}{100}$

☒ many components interpolation slows-down rendering

Full BTF Factorization

[Ruiters & Klein 09] – **sparse tensor decomposition** (K-SVD) into small dictionary tensor and two sparse tensors.

$$BTF(\lambda, r, i, v) \approx \mathcal{L}_{\lambda ij} \times \mathcal{V}_{jvk} \times \mathcal{P}_{kr}$$

$$\mathcal{L}_{\lambda ij} \in \mathbb{R}^{colors \times lights \times D_1}, \quad \mathcal{V}_{jvk} \in \mathbb{R}^{D_1 \times views \times D_2}, \quad \mathbf{P}_{kr} \in \mathbb{R}^{D_2 \times pixels}$$

☑ compact ⇨ high compression $\approx \frac{1}{1000}$

☒ slow compression, long rendering times for high visual quality, no hardware implementation

[Ruiters & Klein 09]

Per-Cluster Factorization

[Muller et al. 03] – combination of iterative clustering of ABRDF data (modified K-Means) and a PCA in each cluster. Eigen-image reconstruction error is used as a distance measure in the clustering process.

$$BTF(r_1, r_2, i, v) \approx \sum_{k=1}^{n_c} \mathbf{u}_{m(r_1,r_2),k}(x,y)\sigma_{m(r_1,r_2),k}\mathbf{v}_{m(r_1,r_2),k}(i,v)$$

☑ high compression $\approx \frac{1}{300}$, good quality

☒ long iterative clustering & compression times

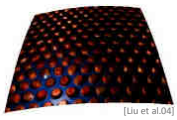
[Muller et al.03]

BTF Space Clustering + Factorisation

[Leung & Malik 01] – 3D textons ⇨ vocabulary of prototype tiny surface patches with associated local geometric and photometric properties. Appearance vectors ⇨ set of linear Gaussian derivative filter outputs.

[Liu et al. 04] – BTF sample ⇨ small number of 4D point apperance functions (textons) multiplied by 2D geometry maps. Textons compressed by linear factorization (SVD).

☑ texton based synthesis, fast HW implemented, high compression



[Liu et al.04]

[Liu et al.04]

BTF Space Clustering + BRDF Model

[Filip & Haindl 05] – ABRDFs clustering (K-means) + polynomial expansion of Lafortune model in cluster centers.

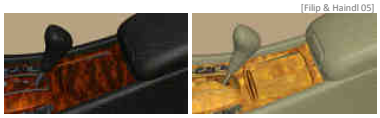
$$\begin{aligned}
\rho(r_1,r_2)_v &= K_{v,1}(I_v(r_1,r_2)) \\
D(r_1,r_2)_{v,X} &= K_{v,2}(I_v(r_1,r_2)) \\
D(r_1,r_2)_{v,Y} &= K_{v,3}(I_v(r_1,r_2)) \\
D(r_1,r_2)_{v,Z} &= K_{v,4}(I_v(r_1,r_2)) \\
n(r_1,r_2)_v &= K_{v,5}(I_v(r_1,r_2))
\end{aligned}$$

$$\mathcal{R}_v(r_1,r_2,i,v) \approx \sum_{j=0}^{n_p} a_{r_3,v,i,j}[\rho_v(r)(\omega_i^T \mathbf{D}_v(r))^{n_v(r)}]^j$$

☑ compression $\approx \frac{1}{200}$, fast fitting of several ABRDFs only

☒ quantization artifacts

[Filip & Haindl 05]



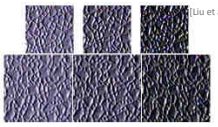
Combining Geometry and Reflectance

[Liu et al. 01] – spare set of small BTF image tiles combined according to range-map to produce novel BTF image of arbitrary size.

☑ arbitrary size BTF synthesis

☒ very slow, low BTF compression

[Liu et al. 01]

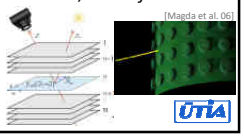


[Magda & Kriegman 06] – separate *geometric information* ⇨ layered volumetric model of material structure, and *reflectance data* ⇨ Lafortune BRDF model.

☑ high compression, easy BTF interpolation

☒ Number of layers and height of the sample ⇨ set explicitly, high grazing angles artifacts

[Magda et al. 06]



Combining Geometry and Reflectance

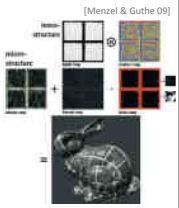
[Menzel & Guthe 09] – **geometric BRDFs** = BTF separation to intuitive and online editable texture maps:

- meso & micro scale geometry ⇨ depth & tangent 2D maps
- light interactions ⇨ Ashikhmin [2006] analytic BRDF model
- segmentation to basis materials

☑ reasonable visual quality, high compression $\approx \frac{1}{1000}$, interactive editing without need of compression

☒ for opaque samples; no sub-surface scattering inter-reflection effects; shadows reconstructed from depth map; BRDF model ⇨ chromatic errors

[Menzel & Guthe 09]



Multilevel Vector Quantization

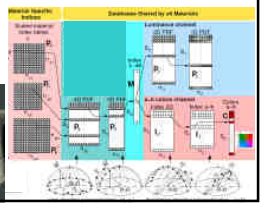
[Havran et al. 10] – BTF multi-level vector quantization algorithm

- BTF parameterized ⇨ set of 1D, 2D, 3D, and 4D data slices
- Multi-level code-books of slices (indices/scales).
- reconstruction ⇨ fast chained indexing in nested code-books

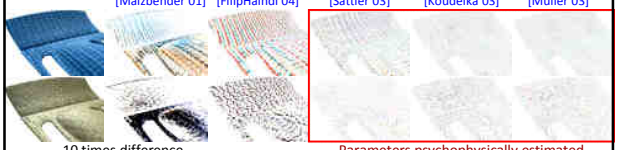
☑ high compression $\approx \frac{1}{700}$, direct importance sampling support, hardware implementation

☒ time demanding generation of optimized codebooks

[Havran et al. 10]



Selected Methods Comparison



	Per-view polynomials [Malzbender 01]	Per-view Lafortune [FilipHaindl 04]	Per-view PCA [Sattler 03]	Entire PCA [Koudelka 03]	Per-cluster PCA [Muller 03]
	10 times difference		Parameters psychophysically estimated		
Compression (average)	$\approx \frac{1}{14}$	$\approx \frac{1}{16}$	$\approx \frac{1}{7}$	$\approx \frac{1}{91}$	$\approx \frac{1}{31}$
Analysis time	≈ 165	≈ 136	≈ 10	≈ 3862	≈ 1098
Synthesis time (25x25 pix.) [s]	≈ 1	≈ 1	≈ 2	≈ 8	≈ 22

Survey [Filip & Haindl 09]

Conclusions on BTF Compression

- BTF ⇨ massive data ⇨ lot of redundancy
- Compression methods
  - **Remove** data **redundancy**
  - Preserve **perceptually important** visual features
  - Allow **random access** for **fast** data reconstruction
- Compression ⇨ reduction of measured data
- Synthesis ⇨ arbitrary BTF sample enlargement
- Synthesis (usually with compression) = **Modelling**
- **BTF data modelling** ⇨ required in practical applications

References

Dimensionality analysis

- Suen, P. and Healey, G, *The Analysis and Recognition of Real-World Textures in Three Dimensions*, IEEE PAMI, 22(5), 491-503, 2000
- Koudelka, M.L. and Magda, S. and Belhumeur, P.N. and Kriegman, D.J, *Acquisition, Compression, and Synthesis of Bidirectional Texture Functions,* Proceedings of Texture 2003, 47—52.
- Filip J., Chantler M.J., Green P.R., Haindl M., *A Psychophysically Validated Metric for Bidirectional Texture Data Reduction*. ACM Transactions on Graphics 27(5) (proceedings of SIGGRAPH Asia 2008), Article 138, December 2008, 11 pp.

Compression methods - Lafortune Model

- E. P. Lafortune, S. Ch. Foo, K. E. Torrance, and D. P. Greenberg. *Non-linear approximation of reflectance functions*. Computer Graphics, 117-126, 1997
- D.K. McAllister, A. Lastra, and W. Heidrich, *Efficient Rendering of Spatial Bi-Directional Reflectance Distribution Functions*, Graphics Hardware, pp. 77-88, 2002.

References

Lafortune Model (continue)

- K. Daubert, H.P.A. Lensch, W. Heidrich, and H.-P. Seidel, *Efficient Cloth Modeling and Rendering*, Proc. Conf. Rendering Techniques, pp. 63-70, 2001.
- J. Meseth, G. Muller, and R. Klein, *Preserving Realism in Real-Time Rendering of Bidirectional Texture Functions*, Proc. OpenSG Symp., pp. 89-96, Apr. 2003.
- J. Filip and M. Haindl, *Non-Linear Reflectance Model for Bidirectional Texture Function Synthesis*, ICPR 2004, vol. 1, pp. 80-83, Aug. 2004.

BRDF Factorisation Methods

- J. Kautz and M.D. McCool. *Interactive rendering with arbitrary BRDFs using separable approximations*. In Proceedings of the 10th Eurographics Workshop on Rendering, pages 281-292, 1999.
- M.D. McCool, J. Ang, and A. Ahmad. *Homomorphic factorization of BRDFs for high-performance rendering*. In E. Fiume, editor, ACM SIGGRAPH 2001, ACM Press, pages 185-194, 2001.
- F. Suykens, K. Berge, A. Lagae, and P. Dutre, *Interactive rendering with bidirectional texture functions*. In Computer Graphics Forum, pages 463-472, 2003.

References

Linear Factorisation Methods

- M. Koudelka, S. Magda, P. Belhumeur, and D. Kriegman, *Acquisition, Compression, and Synthesis of Bidirectional Texture Functions*, TEXTURE 2003, pp. 47-52, Oct. 2003.
- M. Sattler, R. Sarlette, and R. Klein, *Efficient and Realistic Visualization of Cloth*, Proc. Eurographics Symp. Rendering, pp. 167-178, June 2003.
- M. Guthe, G. Muller, M. Schneider, and R. Klein, *BTF-CIELab: A Perceptual Difference Measure for Quality Assessment and Compression of BTFs*, Computer Graphics Forum, vol. 28, no. 1, pp. 101-113, 2009

Multi-modal Linear Factoriazation

- M. Vasilescu and D. Terzopoulos, *Tensor Textures: Multilinear Image-Based Rendering*, Proc. ACM SIGGRAPH, vol. 23, no. 3 pp. 336-342, Aug. 2004.
- H. Wang, G. Wu, L. Shi, Y. Yu, and N. Ahuja, *Out-of-Core Tensor Approximation of Multi-Dimensional Matrices of Visual Data*, ACM Trans. Graphics, vol. 24, no. 3, pp. 527-535, July 2005.
- R. Ruiters and R. Klein, *BTF Compression via Sparse Tensor Decomposition*, Computer Graphics Forum, vol. 28, no. 4, pp. 1182-1188, 2009

References

Polynomial Texture Maps

- T. Malzbender, D. Gelb, and H. Wolters, *Polynomial Texture Maps*, Proc. ACM SIGGRAPH, pp. 519-528, 2001.

Clustering + Linear Factorisation

- G. Müller, J. Meseth, and R. Klein. *Compression and real-time rendering of measured BTFs using local PCA*. To appear in proceedings of Vision, Modeling and Visualisation 2003, November 2003.
- T. Leung and J. Malik, *Representing and Recognizing the Visual Appearance of Materials Using Three-Dimensional Textons*, Int'l J. Computer Vision, vol. 43, no. 1, pp. 29-44, 2001.
- X. Liu, J. Zhang, X. Tong, B. Guo, and H.-Y. Shum. *Synthesis and rendering of bidirectional texture functions on arbitrary surfaces*. IEEE Transactions on Visualization and Computer Graphics, 10(3):278-289, June 2004.

Vector Quantization

- Havran V., Filip J., Myzskowski K., *Bidirectional Texture Function Compression based on Multi-Level Vector Quantization*. Computer Graphics Forum, vol. 29, no. 1, pp. 175-190, March 2010

References

Combining Geometry and Reflectance

- X. Liu, Y. Yu, and H.Y. Shum, *Synthesizing Bidirectional Texture Functions for Real-World Surfaces*, Proc. ACM SIGGRAPH, pp. 97-106, 2001.
- S. Magda and D. Kriegman, *Reconstruction of Volumetric Surface Textures for Real-Time Rendering*, Proc. 17th Eurographics Workshop Rendering, pp. 19-29, June 2006.
- N. Menzel and M. Guthe, *g-BRDFs: An Intuitive and Editable BTF Representation*, Computer Graphics Forum, vol. 28, no. 8, pp. 2189 - 2200, 2009

Combining Geometry and Reflectance

- G. Müller, R. Sarlette, and R. Klein, *Data-driven Local Coordinate Systems for Image-Based Rendering*, In: Computer Graphics Forum, 25:3, 2006