Reconstruction of Smooth 3D Color Functions from Keypoints: Application to Lossy Compression and Examplar-Based Generation of Color LUTs

D. Tschumperlé C. Porquet A. Mahboubi

Normandie Univ, UNICAEN, ENSICAEN, CNRS GREYC,
F-14000 Caen, France

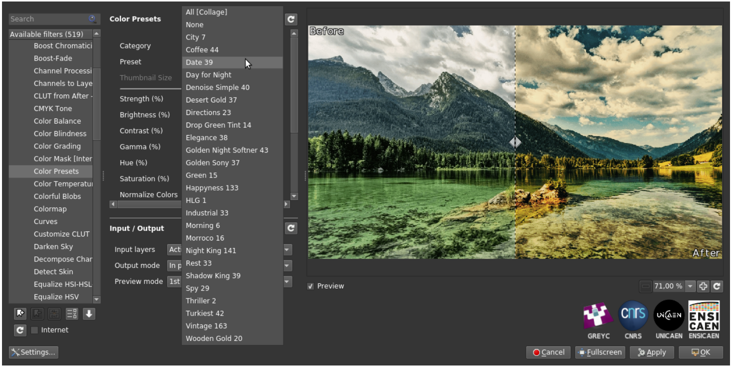
June 2020

Context

A Full-Featured Open-Source Framework for Image Processing

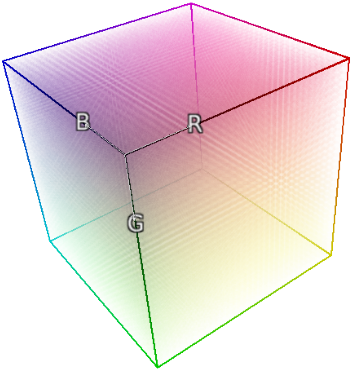
GREYC's **M**agic for **I**mage **C**omputing

https://gmic.eu



- **Application context:**
  Provide *various colorimetric transformations* available in our open-source framework for image processing.

What is a *CLUT* (*Color Look-Up Table*) ?



(a) *CLUT* **F** : *RGB* → *RGB*,
visualized in 3*D*

(b) Original color image

(c) Image after transformation **F**

Examples of of *CLUT*-based transformations

Original image

“60’s”

“Color Negative”

“Orange Tone”

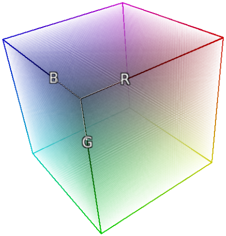
“Ilford Delta 3200”

“Backlight Filter”

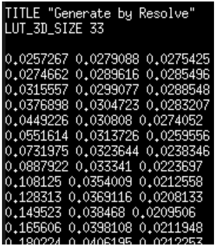
“Bleach Bypass”

“Late Sunset”

Standard ways of storing a *CLUT*



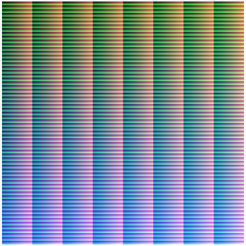
a) a *CLUT* is a $3D$ dense color volume



```
TITLE "Generate by Resolve"
LUT_3D_SIZE 33

0.0257267 0.0279088 0.0275425
0.0274662 0.0289616 0.0285496
0.0315557 0.0299077 0.0288548
0.0376898 0.0304723 0.0283207
0.0449226 0.030808 0.0274052
0.0551614 0.0313726 0.0259556
0.0731975 0.0323644 0.0238346
0.0887922 0.033341 0.0223697
0.108125 0.0354009 0.0212558
0.128313 0.0369116 0.0208133
0.149523 0.038468 0.0209506
0.165606 0.0398108 0.0211948
```

b) Storage as a `.cube`



c) Storage as a `.png`

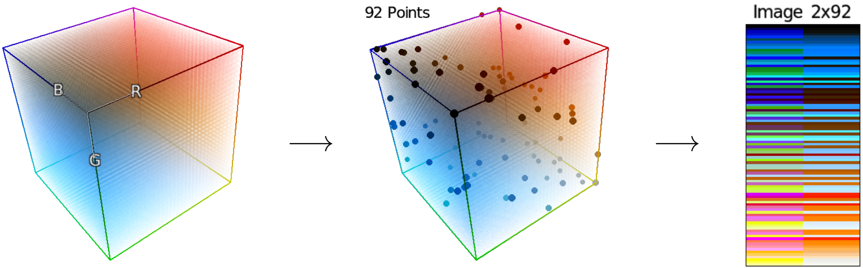
In both cases, lossless compression, but restricted to small sizes:

1. `.cube` file: *ASCII* zipped format (*CLUT* $64^3 \approx$ 1 Mo)
2. `.png` file: $2D$ image (*CLUT* $64^3 \approx$ 64 to 100 Ko)

$\Rightarrow$ **Issue:** Promote a large-scale distribution of *CLUT*s (**500+**), by limiting the number of parameters as much as possible.

Our approach: CLUT compression

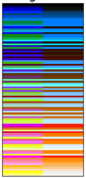
Compression: Let $\mathbf{F} : RGB \rightarrow RGB$ be a $3D$ *CLUT*.
We generate $\mathcal{K}$, a smaller representation based on the storage of a set of color keypoints.



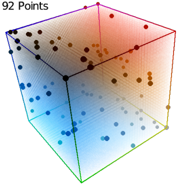
Original CLUT $\mathbf{F}$ → Determination of $3D$ color keypoints $\mathcal{K}$ → Storage in compressed form

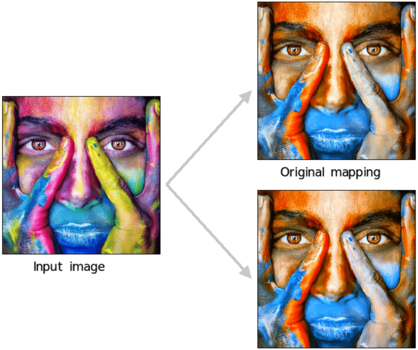
Our approach: CLUT decompression

Decompression: A $3D$ interpolation based on anisotropic diffusion *PDEs* is applied to $\mathcal{K}$ in order to generate a reconstructed CLUT $\widetilde{\mathbf{F}}$ visually close to $\mathbf{F}$.



$\rightarrow$





Stockage in compressed form

Decompressed CLUT $\widetilde{\mathbf{F}}$

No visible perceptual differences between the two transformations

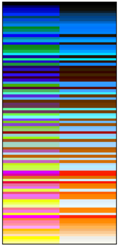
Reconstruction principles (1/3)

Let $\mathbf{F} : RGB \rightarrow RGB$ be a $3D$ *CLUT*.
It is assumed that its sparse representation is known.

$$\mathcal{K} = \{\mathbf{K}_k \in RGB \times RGB \mid k = 1 \dots N\}$$

i.e the color keypoints N, located in the *RGB* cube.

Image 2x92



The k^{th} keypoint of $\mathcal{K}$ is defined by vecteur

$$\mathbf{K}_k = (\mathbf{X}_k, \mathbf{C}_k) = (x_k, y_k, z_k, R_k, G_k, B_k),$$

where $\mathbf{X}_k = (x_k, y_k, z_k)$ is the $3D$ keypoint position in the *RGB* cube, and $\mathbf{C}_k = (R_k, G_k, B_k)$ its associated color.

Reconstruction principles (2/3)

We propagate/average the colors $\mathbf{C}_k$ of the keypoints in the whole *RGB* domain throgh an **anisotropic diffusion** process

- Let $d_{\mathcal{K}} : RGB \rightarrow \mathbb{R}^+$, the distance function to the set of keypoints $\mathcal{K}$:

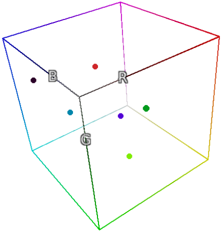
$$\forall \mathbf{X} \in RGB, \qquad d_{\mathcal{K}(\mathbf{X})} = \inf_{k \in 0 \dots N} \| \mathbf{X} - \mathbf{X}_k \|$$

- *CLUT* $\mathbf{F}$ is reconstructed by solving the following anisotropic diffusion *PDE* :

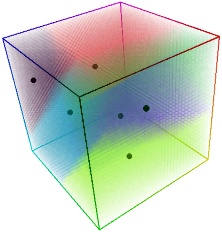
$$\forall \mathbf{X} \in RGB, \qquad \frac{\partial \mathbf{F}}{\partial t}(\mathbf{X}) = m_{(\mathbf{X})} \, \frac{\partial^2 \mathbf{F}}{\partial \eta^2}(\mathbf{X})$$

$$\text{where} \quad \eta = \frac{\nabla d_{\mathcal{K}(\mathbf{X})}}{\| \nabla d_{\mathcal{K}(\mathbf{X})} \|} \quad \text{and} \quad m_{(\mathbf{X})} = \begin{cases} 0 & \text{if} \quad \exists k, \quad \mathbf{X} = \mathbf{X}_k \\ 1 & \text{otherwise} \end{cases}$$

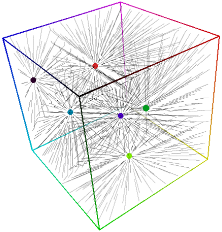
Reconstruction principles (3/3)



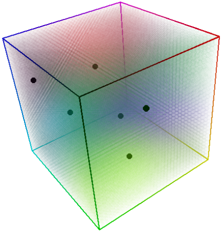
(a) Set $\mathcal{K}$ of known keypoints



(b) Initial state $\mathbf{F}_{t=0}$ (*Voronoï 3D smoothed*)



(c) Diffusion orientations η



(d) State at convergence (*PDE* Solution)

Spatial discretization (1/2)

$d_{\mathcal{K}}$ is not derivable on its local maxima.
We propose the following numerical scheme for discretization:

$$\nabla d_{\mathcal{K}(\mathbf{x})} = \begin{pmatrix} \mathrm{maxabs}(\partial_x^{\mathrm{for}} d_{\mathcal{K}}, \partial_x^{\mathrm{back}} d_{\mathcal{K}}) \\ \mathrm{maxabs}(\partial_y^{\mathrm{for}} d_{\mathcal{K}}, \partial_y^{\mathrm{back}} d_{\mathcal{K}}) \\ \mathrm{maxabs}(\partial_z^{\mathrm{for}} d_{\mathcal{K}}, \partial_z^{\mathrm{back}} d_{\mathcal{K}}) \end{pmatrix}$$

where

$$\mathrm{maxabs}(a,b) = \begin{cases} a & \text{if } |a| > |b| \\ b & \text{otherwise} \end{cases}$$

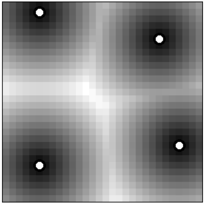
and

$$\partial_x^{\mathrm{for}} d_{\mathcal{K}} = d_{\mathcal{K}(x+1,y,z)} - d_{\mathcal{K}(x,y,z)}$$

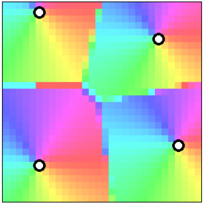
$$\partial_x^{\mathrm{back}} d_{\mathcal{K}} = d_{\mathcal{K}(x,y,z)} - d_{\mathcal{K}(x-1,y,z)}$$

(Discrete *forward*/*backward* first derivative approximations).

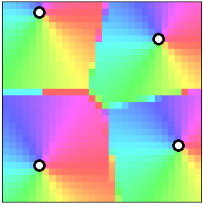
Spatial discretization (2/2)



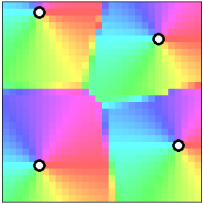
(a) Keypoints and distance function $d_{\mathcal{K}}$



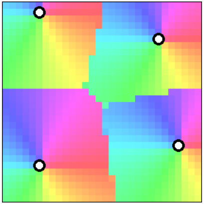
(b) Estimation of η using *forward* scheme $\partial^{\,\text{for}} d_{\mathcal{K}}$



(c) Estimation of η using *backward* scheme $\partial^{\,\text{back}} d_{\mathcal{K}}$



(d) Estimation of η using *centred* scheme $\frac{1}{2}(\partial^{\,\text{for}} d_{\mathcal{K}} + \partial^{\,\text{back}} d_{\mathcal{K}})$



(e) Estimation of η using proposed scheme

Temporal discretization

For the sake of algorithmic efficiency, we discrete the *PDE* by a *semi-implicit* scheme:

$$\frac{\mathbf{F}^{t+dt} - \mathbf{F}^{t}}{dt} = m_{(\mathbf{X})} \left[\mathbf{F}^{t}_{(\mathbf{X}+\eta)} + \mathbf{F}^{t}_{(\mathbf{X}-\eta)} - 2\,\mathbf{F}^{t+dt}_{(\mathbf{X})} \right]$$

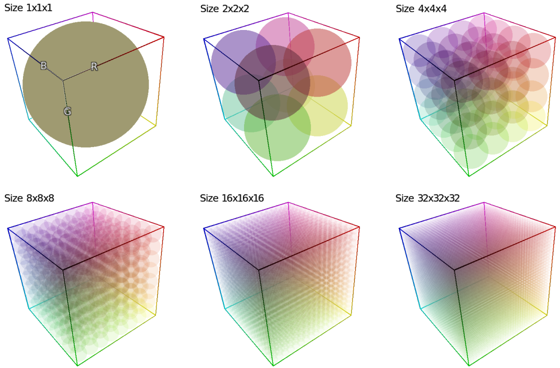
By choosing dt appropriately, we simplify the scheme by:

$$\begin{cases} \mathbf{F}^{t+dt}_{(\mathbf{X})} = \mathbf{F}^{t}_{(\mathbf{X})} & \text{if } m_{(\mathbf{X})} = 0 \\ \mathbf{F}^{t+dt}_{(\mathbf{X})} = \frac{1}{2} \left[\mathbf{F}^{t}_{(\mathbf{X}+\eta)} + \mathbf{F}^{t}_{(\mathbf{X}-\eta)} \right] & \text{otherwise} \end{cases}$$

where $\mathbf{F}^{t}_{(\mathbf{X}+\eta)}$ and $\mathbf{F}^{t}_{(\mathbf{X}-\eta)}$ are accurately estimated using tricubic spatial interpolation.

Multi-scale resolution

- Speed-up of convergence through **multi-scale resolution**.

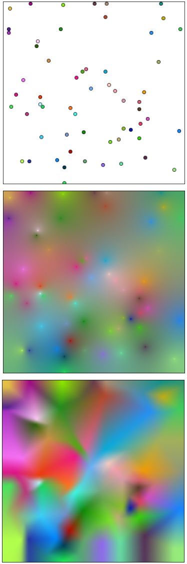


→ Reduction of the number of required iterations per scale (≈ 20).

→ Total algorithmic complexity for a size r^3 : $O(log_2(r)\ r^3)$.

→ Reconstruction of a 64^3 *CLUT* in less than 1s.

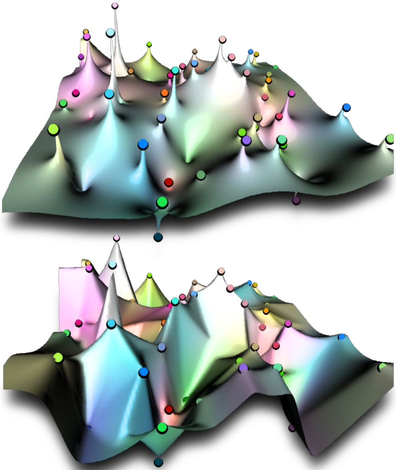
Isotropic/Anisotropic comparison



(a)

(b)

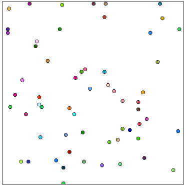
(c)



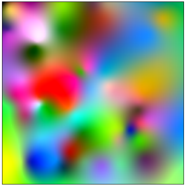
(a) Color Keypoints $\mathcal{K}$

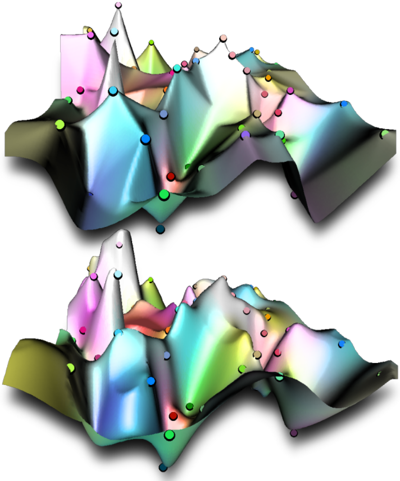
(b) **Isotropic** reconstruction, in $O(log_2(r)\ r^3)$

(c) **Anisotropic** reconstruction, in $O(log_2(r)\ r^3)$

(a) 

(b) 

(c) 



(a) Color Keypoints $\mathcal{K}$

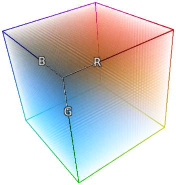
(b) **Anisotropic** reconstruction, in $O(log_2(r)\ r^3)$

(c) **RBFs** reconstruction, in $O(N^3 + N\ r^3)$.

Compression: Generation of keypoints

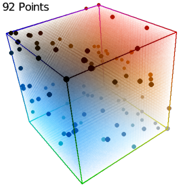
Let $\mathbf{F}$ be the *CLUT* to be compressed. The compression:

- Generates a set $\mathcal{K}$ of N keypoints representing $\mathbf{F}$.
- $\widetilde{\mathbf{F}}_N$ the reconstructed *CLUT* from $\mathcal{K}$ must be close enough to $\mathbf{F}$.
  Two quality criteria for reconstruction:
  - $\Delta_{\text{max}} = 8$, maximum reconstruction error allowed,
  - $\Delta_{\text{moy}} = 2$, average reconstruction error for the entire *CLUT*



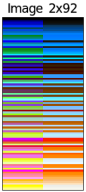
Original *CLUT* $\mathbf{F}$

$\longrightarrow$



Determination of 3*D* color keypoints $\mathcal{K}$

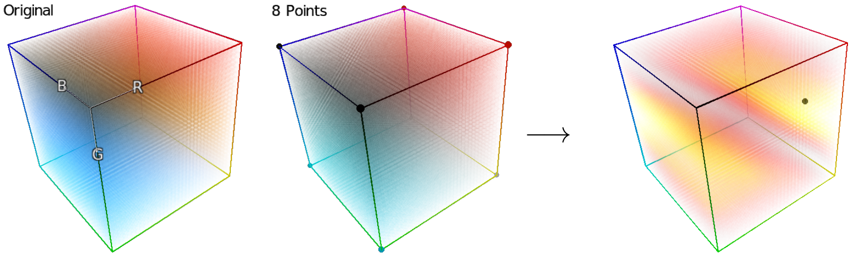
$\longrightarrow$



Storage in compressed form

Compression: Generation of keypoints

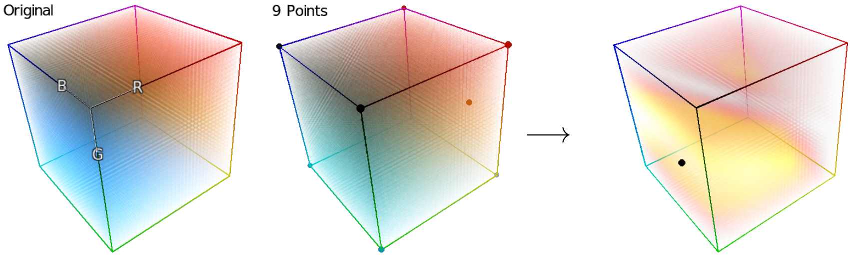
1. Initialization of $\mathcal{K} = \{(\mathbf{X}_k, \mathbf{F}_{(\mathbf{X}_k)} \mid k = 1 \ldots 8\}$ (the 8 vertices of the cube).



$\Rightarrow$ Calculation of the L_2 reconstruction error: $E_{N(\mathbf{X})} = \|\mathbf{F}_{(\mathbf{X})} - \widetilde{\mathbf{F}}_{N(\mathbf{X})}\|$.

Compression: Generation of keypoints

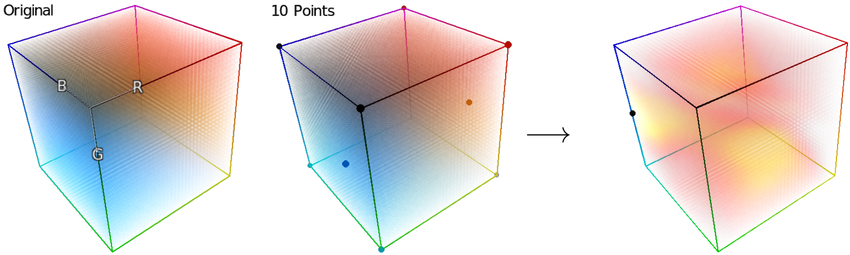
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\mathrm{max}} > \Delta_{\mathrm{max}}$ or $E_{\mathrm{moy}} > \Delta_{\mathrm{moy}}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

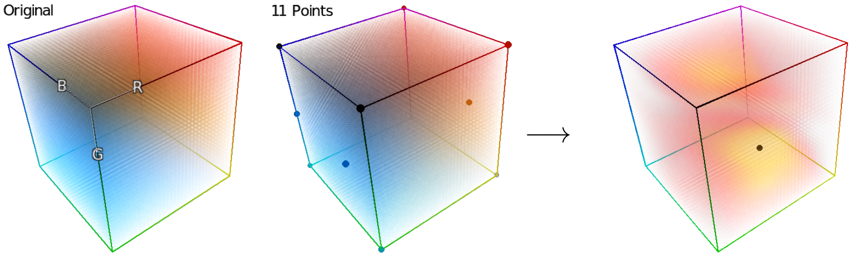
Compression: Generation of keypoints

2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\text{max}} > \Delta_{\text{max}}$ or $E_{\text{moy}} > \Delta_{\text{moy}}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

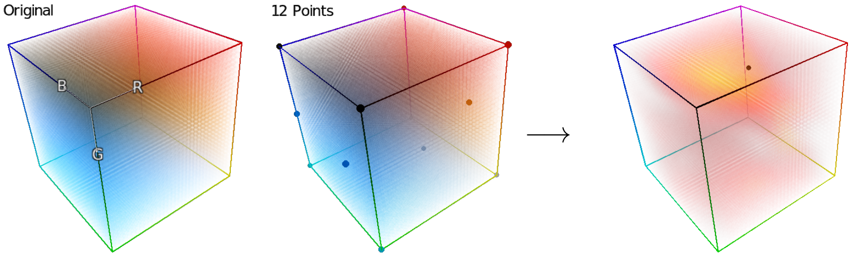
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\max} > \Delta_{\max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

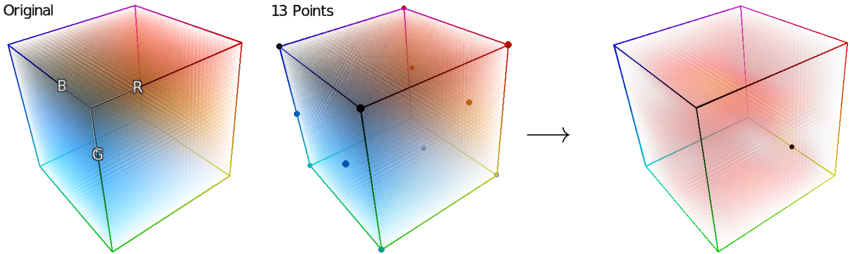
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

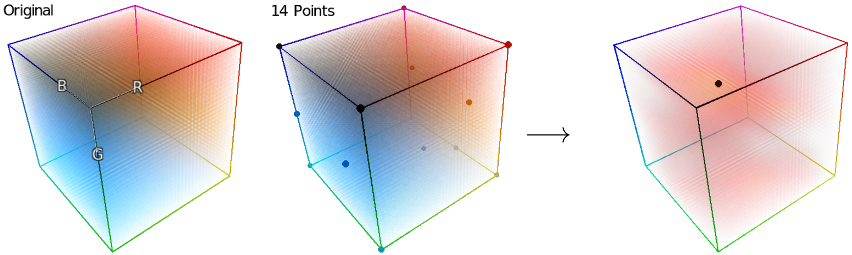
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

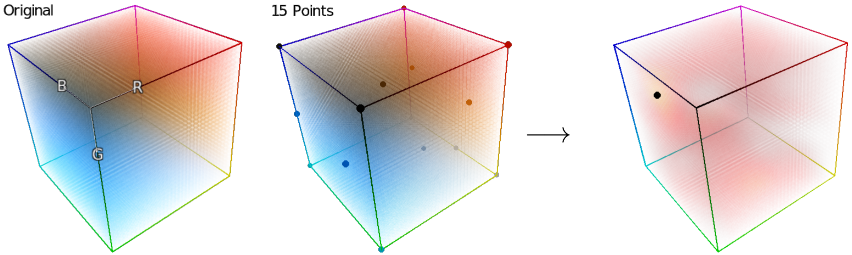
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

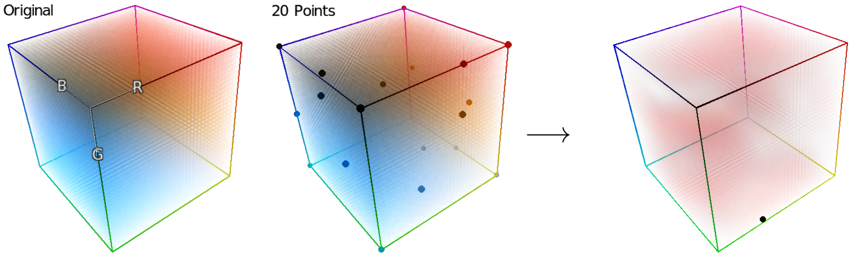
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

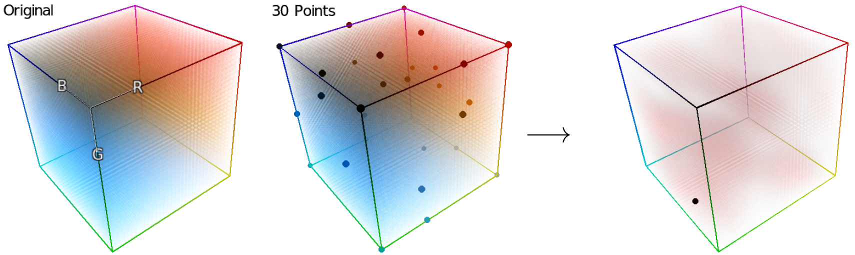
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

Compression: Generation of keypoints

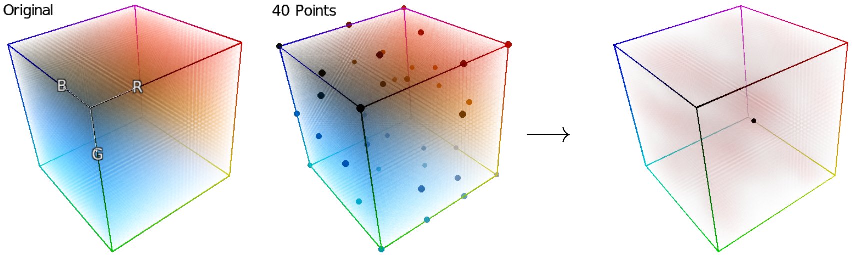
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\max} > \Delta_{\max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

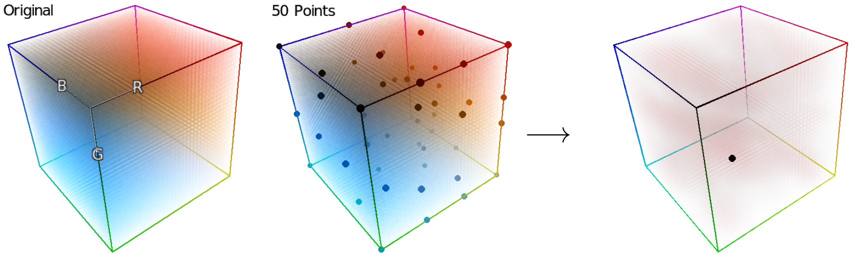
Compression: Generation of keypoints

2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\text{max}} > \Delta_{\text{max}}$ or $E_{\text{moy}} > \Delta_{\text{moy}}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

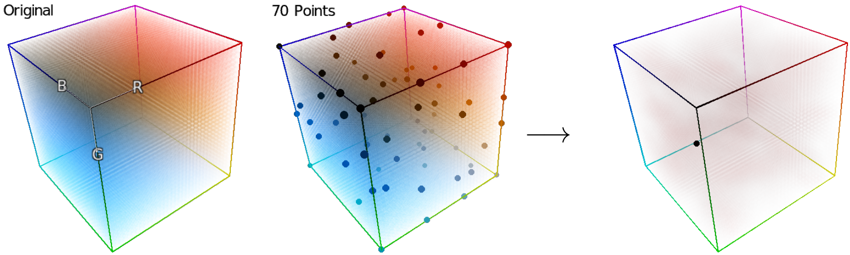
2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{max} > \Delta_{max}$ or $E_{moy} > \Delta_{moy}$.



$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

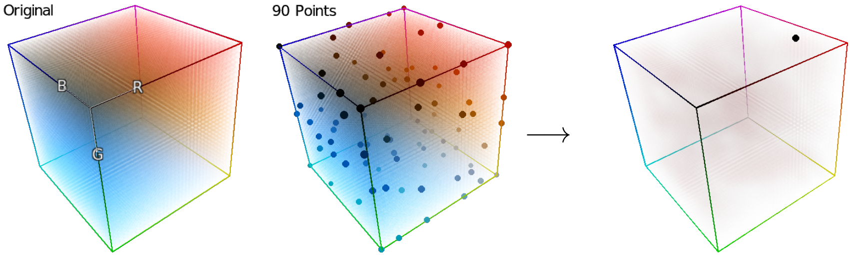
Compression: Generation of keypoints

2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\text{max}} > \Delta_{\text{max}}$ or $E_{\text{moy}} > \Delta_{\text{moy}}$.



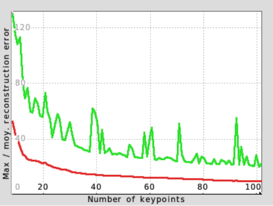
$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

2. Iterative addition into $\mathcal{K}$ of the keypoints with maximum reconstruction error, while $E_{\text{max}} > \Delta_{\text{max}}$ or $E_{\text{moy}} > \Delta_{\text{moy}}$.

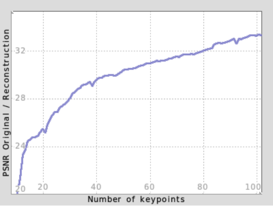


$\Rightarrow$ Calculation of the reconstruction error: $E_{N(\mathbf{x})} = \|\mathbf{F}_{(\mathbf{x})} - \widetilde{\mathbf{F}}_{N(\mathbf{x})}\|$.

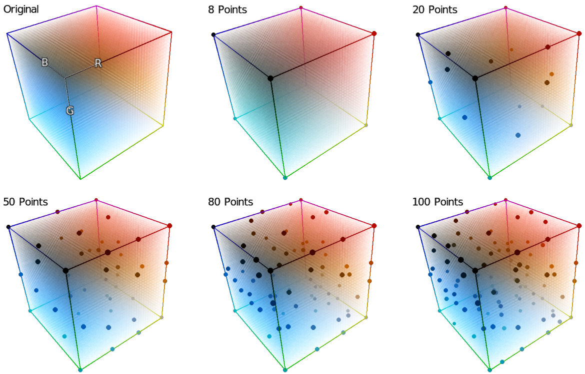
Adding keypoints for *CLUT* compression



(a) max/average error



(b) PSNR evolution



(c) Iterative addition of keypoints