
Laboratorio di Immagini

Esercitazione 8:

Spazi Colore

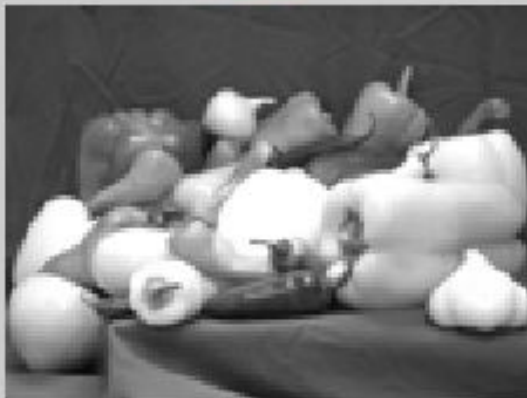
Sir. Mauro Zucchelli
19/05/2016

Colori RGB

- In MATLAB di default quando viene letta un immagine a colori viene caricata nello spazio RGB
 - Red
 - Green
 - Blue
 - Questo spazio colore, derivante dalla fisica, potrebbe non essere il migliore per elaborare le immagini
-

RGB: canali

RED



GREEN



BLUE



COLOR

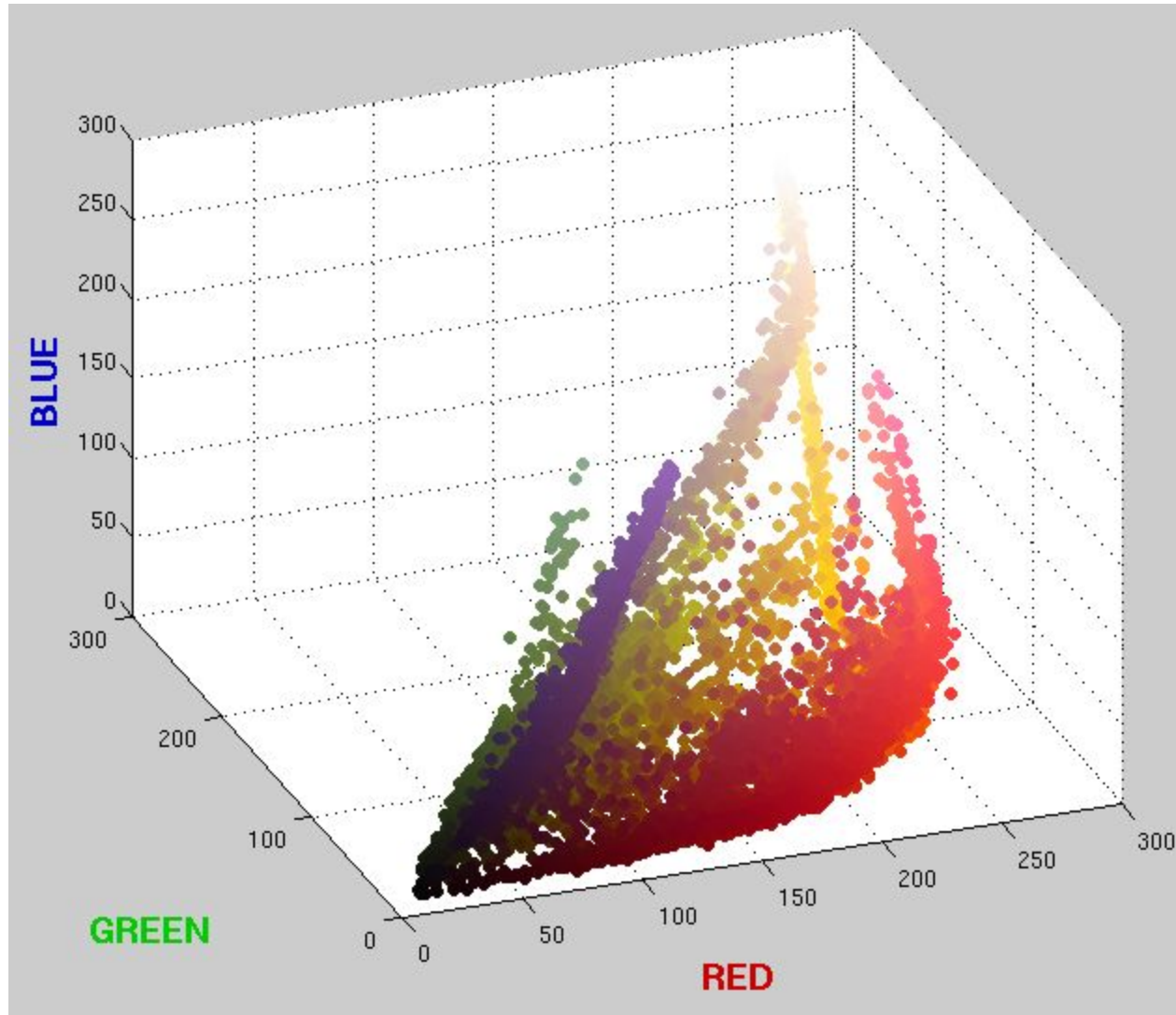


RGB: canali

```
img = imread('peppers.png');
img = imresize(img,0.25);

rosso = img(:,:,1);
verde = img(:,:,2);
blu = img(:,:,3);
figure('Name','RGB');
subplot(2,2,1);
imshow(rosso);
title('RED','FontSize',14,'fontweight','bold','Color',[0.8,0,0])
subplot(2,2,2);
imshow(verde);
title('GREEN','FontSize',14,'fontweight','bold','Color',[0,0.8,0]);
subplot(2,2,3);
imshow(blu);
title('BLUE','FontSize',14,'fontweight','bold','Color',[0,0,0.8])
subplot(2,2,4);
imshow(img);
title('COLOR','FontSize',14,'fontweight','bold')
```

RGB: spazio colore



RGB: spazio colore

```
[x,y,c] = size(img);  
  
img_rgb = double(reshape(img,[x*y,c]));  
figure;  
scatter3(img_rgb(:,1), img_rgb(:,2), img_rgb(:,3), [],img_rgb/256.0, 'filled')  
xlabel('RED','FontSize',14,'fontweight','bold','Color',[0.8,0,0]);  
ylabel('GREEN','FontSize',14,'fontweight','bold','Color',[0,0.8,0]);  
zlabel('BLUE','FontSize',14,'fontweight','bold','Color',[0,0,0.8]);
```

RGB: K-means 5 cluster

ORIGINAL



SEGMENTATION



RGB: K-means 5 cluster

```
k = 5;
```

```
M = zeros(k,3);
```

```
M(:,1) = prctile(img_rgb(:,1),linspace(5,95,k));
```

```
M(:,2) = prctile(img_rgb(:,2),linspace(5,95,k));
```

```
M(:,3) = prctile(img_rgb(:,3),linspace(5,95,k));
```

```
labels= my_kmeans(img_rgb, M,k);
```

```
labels = reshape(labels,[x,y,3]);
```

```
figure
```

```
subplot(1,2,1)
```

```
imshow(img)
```

```
title('ORIGINAL','FontSize',14,'fontweight','bold')
```

```
subplot(1,2,2)
```

```
imshow(labels)
```

```
title('SEGMENTATION','FontSize',14,'fontweight','bold')
```

Colori RGB

- Provate ad adattare il codice per testare il K-MEANS sullo spazio RGB per
 - lena.jpg
 - penguin.jpg
 - O la vostra immagine preferita
-

Colori YCbCr

- RGB è un formato molto ridondante
 - **YCbCr** è uno spazio colore molto usato nella fotografia digitale e nei video
 - 3 canali colore:
 - **Y** o “Lume” contiene l’informazione di luminanza (una sorta di versione grayscale dell’immagine)
 - **Cb** contiene la cromaticanza blu
 - **Cr** contiene la cromaticanza rossa
-

Colori YCbCr

- $Y = 16 + 1/256 * (66 * R + 129 * G + 25 * B)$
 - $Cb = 128 + 1/256 * (- 38 * R - 75 * G + 112 * B)$
 - $Cr = 128 + 1/256 * (112 * R - 94 * G - 18 * B)$
-

Colori YCbCr

- Data l'immagine peppers.png provate a convertirla in YCbCr con il comando "rgb2ycbcr" e visualizzate
 - I tre canali dello spazio
 - Lo spazio colore in un grafico 3D (*usate come colore dei pallini il codice RGB*)
 - Il K-means con 5 cluster (*ricordate di riconvertire le labels con il comando ycbcr2rgb!*)
-

Colori YCbCr

Y



Cb



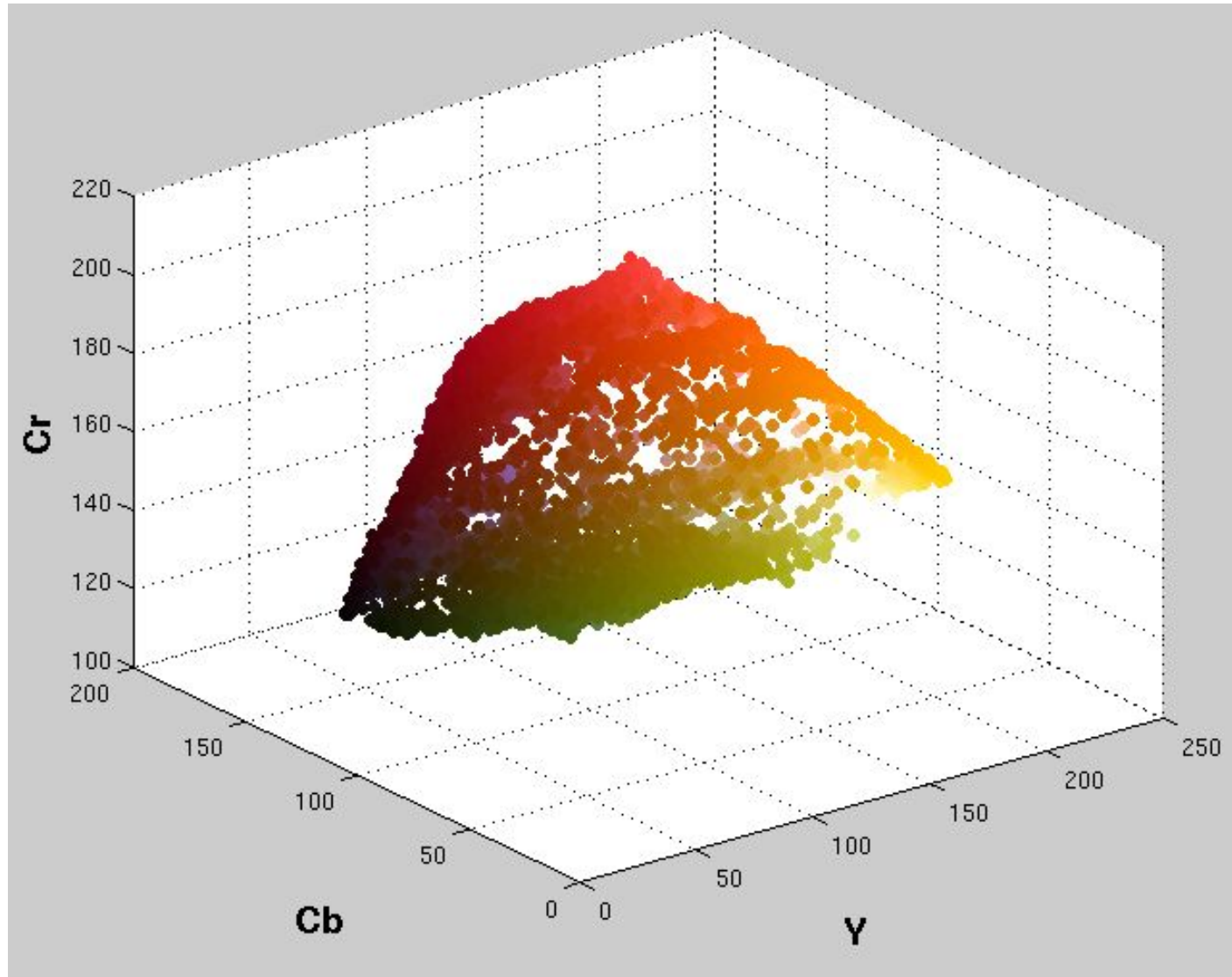
Cr



COLOR



Colori YCbCr

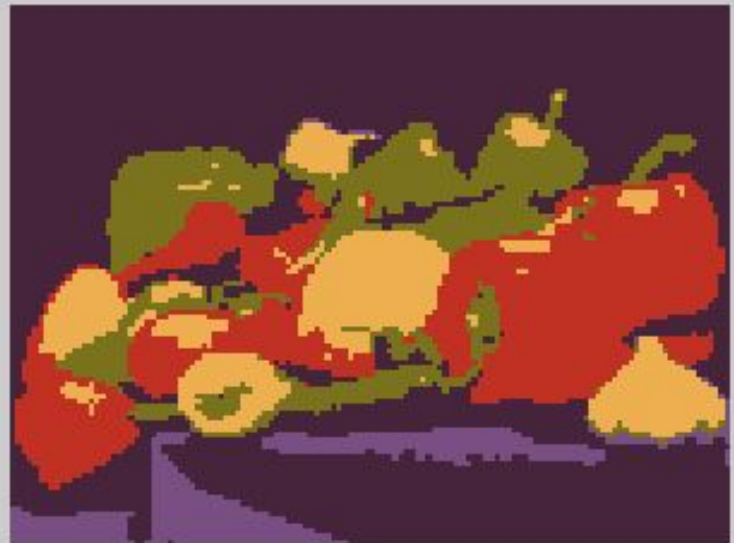


Colori YCbCr

ORIGINAL



SEGMENTATION

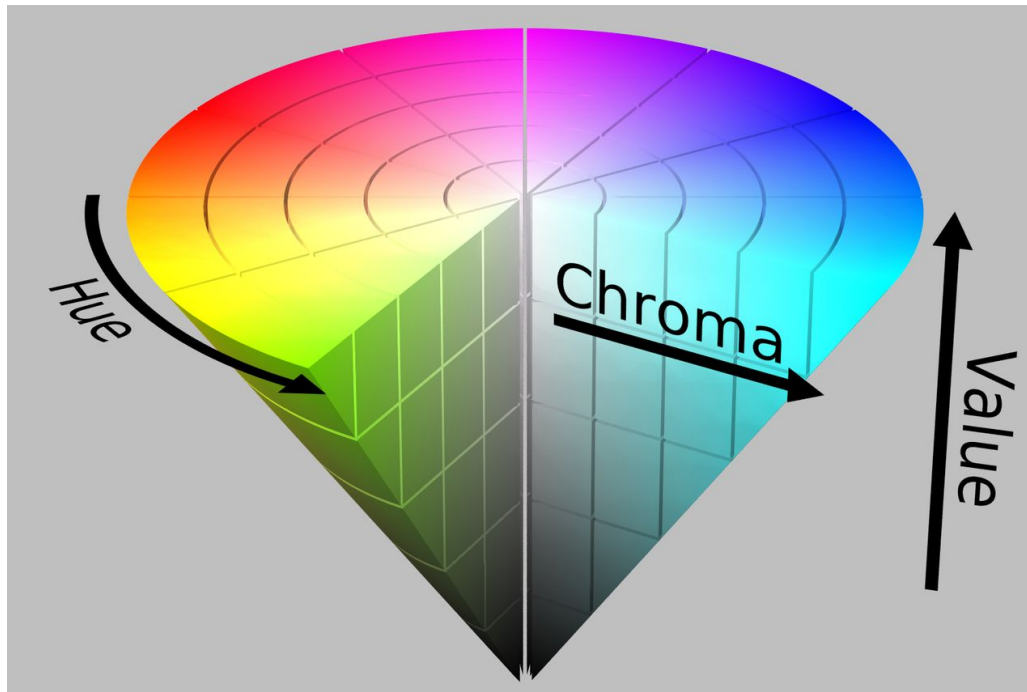


Colori YCbCr

- Provate ad adattare il codice per testare il K-MEANS sullo spazio YCbCr per
 - lena.jpg
 - penguin.jpg
 - O la vostra immagine preferita
-

Colori HSV

- **Hue Saturation Value (HSV)**, in inglese "tonalità, saturazione e valore", indica sia un metodo additivo di composizione dei colori. [Cit. Wikipedia]



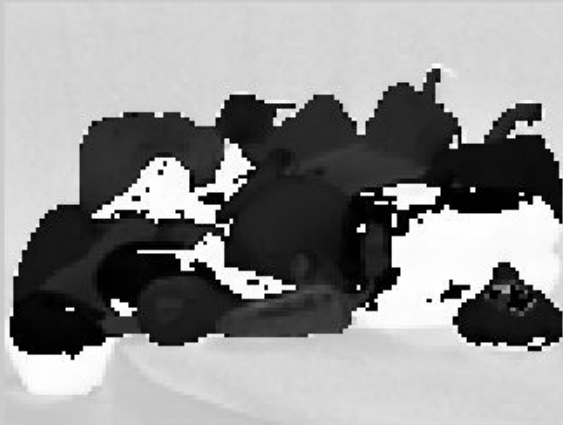
Colori HSV

- Data l'immagine peppers.png provate a convertirla in HSV con il comando "rgb2hsv" e visualizzate
 - I tre canali dello spazio
 - Lo spazio colore in un grafico 3D
 - Il K-means con 5 cluster (*ricordate di riconvertire le labels con il comando hsv2rgb!*)

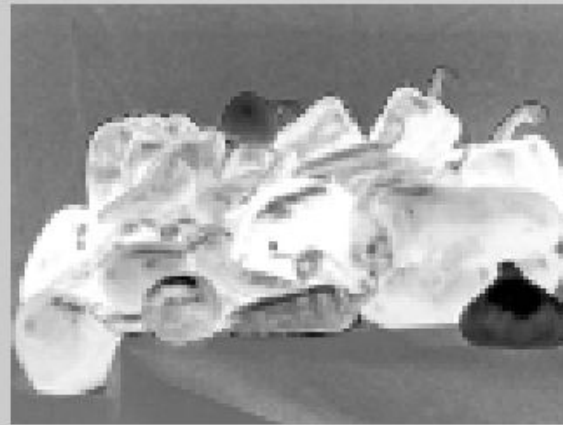
ATTENZIONE: hsv è definito tra 0 e 1 e non tra 0 e 255!

Colori HSV

Hue



Saturation



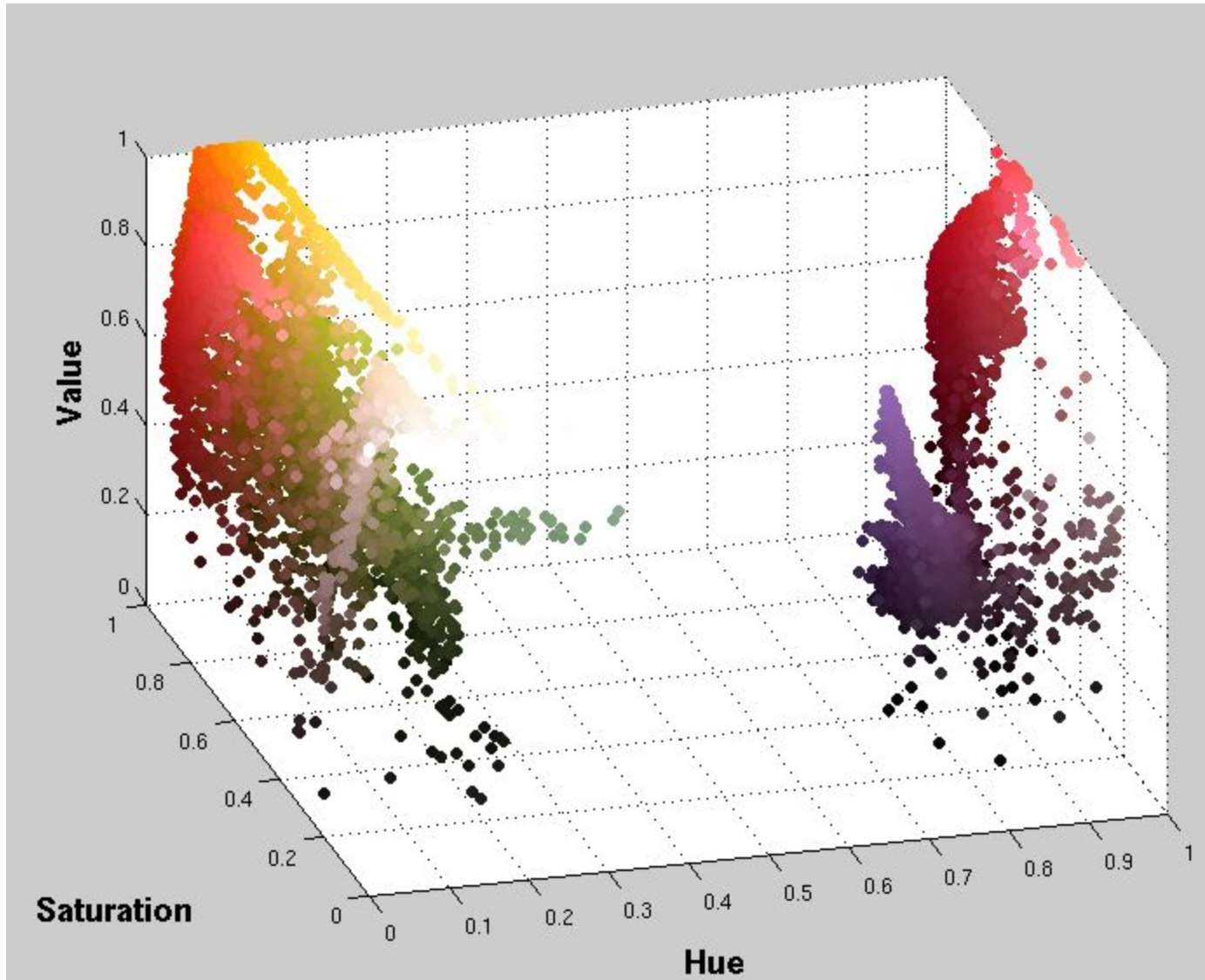
Value



COLOR



Colori HSV



Colori HSV

ORIGINAL



SEGMENTATION



Colori HSV

- Nello spazio HSV la “tinta” (Hue) è definita in maniera ciclica
 - Questo significa che nel canale **H** il colore corrispondente a valore 1.0 è molto vicino a quello a colore 0.0
 - La distanza Euclidea non è una buona scelta in questo caso
-

Colori HSV (2)

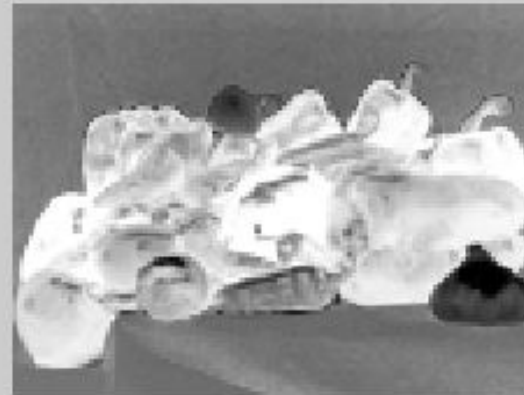
- Trucchetto: possiamo cambiare il canale **H** usando una semplice normalizzazione:
 - $\text{img2}(:,1) = \sin(\text{img2}(:,1)*\pi);$
 - Sfruttiamo il fatto che $\sin(\pi) = \sin(0)$
 - Ovviamente dobbiamo riconvertire anche le labels:
 - $\text{labels}(:,1) = \text{asin}(\text{labels}(:,1)/\pi);$
-

Colori HSV (2)

Hue



Saturation



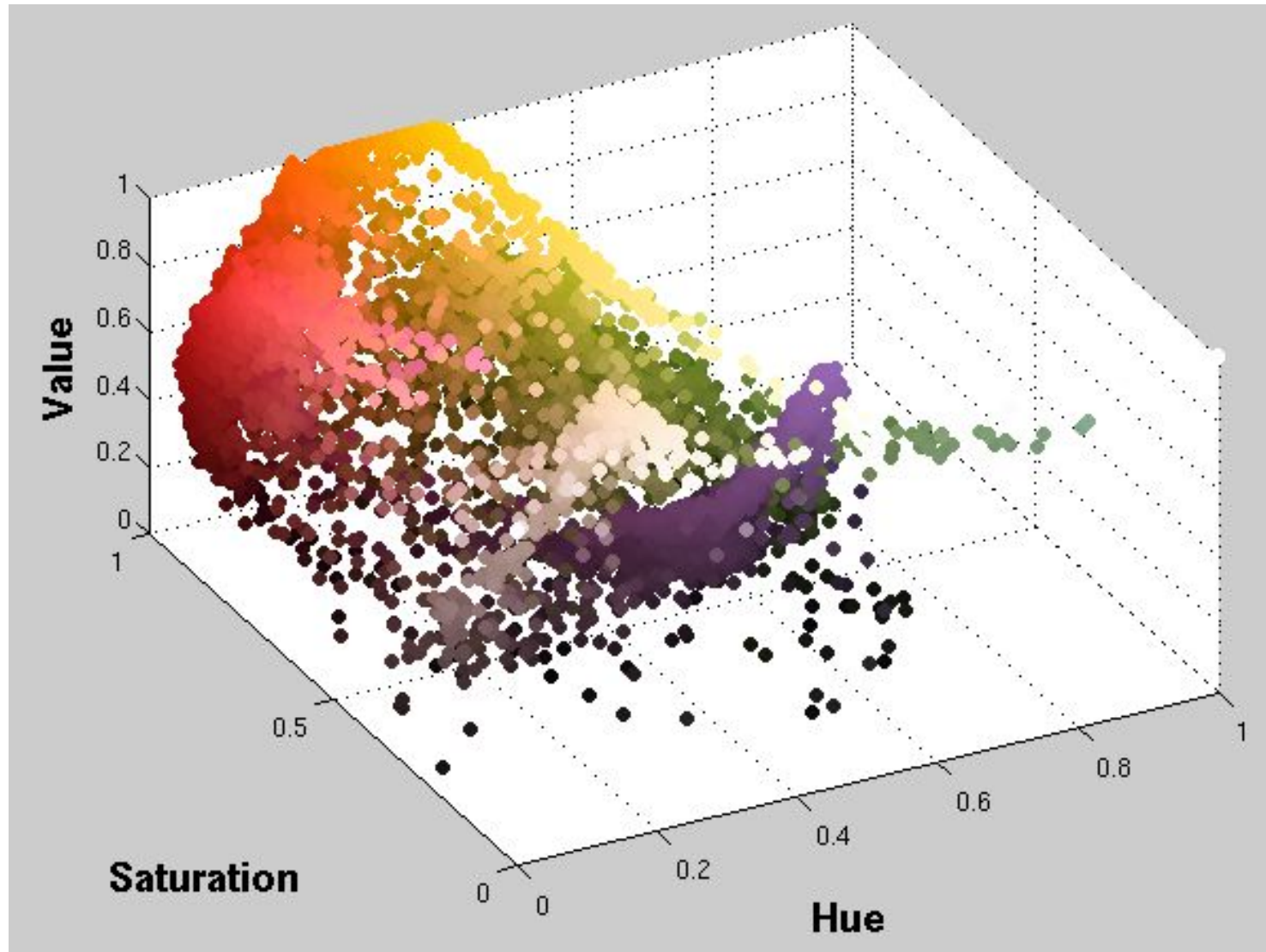
Value



COLOR



Colori HSV (2)



Colori HSV (2)

ORIGINAL



SEGMENTATION



Colori LAB

- **Idea:** serviva uno spazio colore “comune” per gestire le immagini che fosse indipendente dal dispositivo usato per la visualizzazione (monitor o stampante)
 - È stato progettato per approssimare il **sistema visivo umano** con una componente luminosità e uno spazio colore uniforme
-

Colori LAB

- Adattato da WIKIPEDIA:
 - Lo **Spazio colore Lab** o **CIELAB** o **CIE 1976** (**L^*** , **a^*** , **b^***) è uno spazio colore con la dimensione **L** per la luminosità e **a** e **b** per le dimensioni del colore
 - È basato sulle coordinate dello spazio colore non lineare compresso CIE XYZ
-

Colori LAB

- La **luminosità** è calcolata usando la radice cubica della luminanza relativa
 - Lab include tutti i colori percepibili ed è **indipendente** dal dispositivo che li rappresenta
-

Colori LAB

- Data l'immagine peppers.png provate a convertirla in LAB e visualizzare:
 - i tre canali dello spazio
 - Lo spazio colore in un grafico 3D
 - Il K-means con 5 cluster
-

Colori LAB in MATLAB

- Da RGB a LAB:

```
colorTransform = makecform('srgb2lab');
```

```
lab = applycform(img, colorTransform);
```

- Da LAB a RGB:

```
colorTransform = makecform('lab2srgb');
```

```
labels = applycform(labels, colorTransform);
```

Colori LAB

L



A



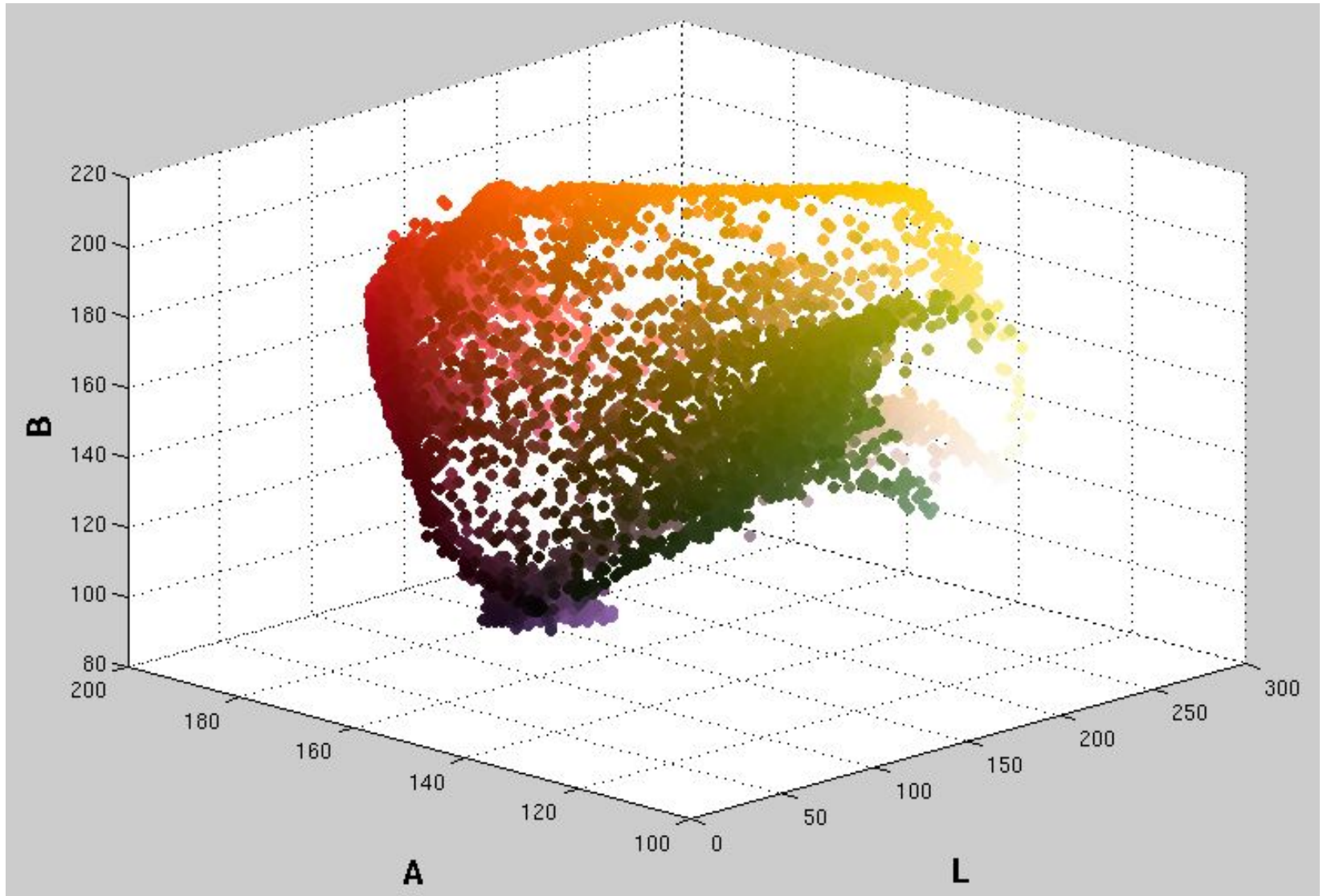
B



COLOR



Colori LAB

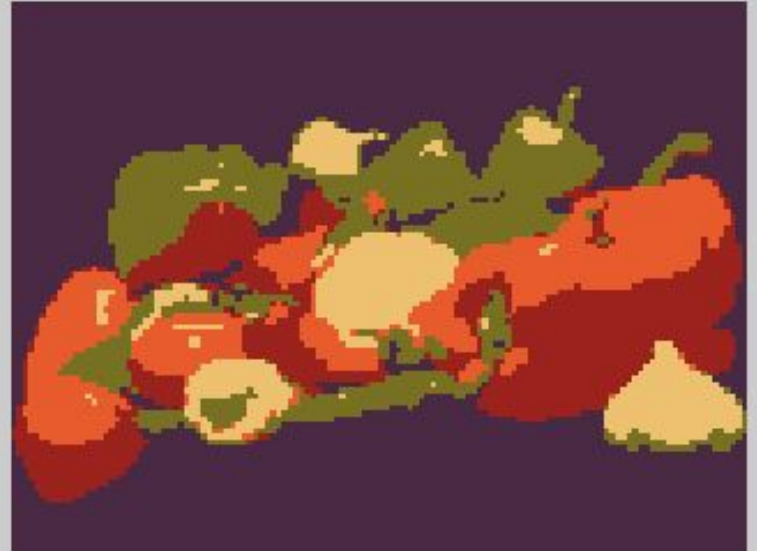


Colori LAB

ORIGINAL



SEGMENTATION



FINE!

The End
