A Model of Saliency-Based Visual Attention for Rapid Scene Analysis

Itti, L., Koch, C., Niebur, E.

Presented by Russell Reinhart
CS 674, Fall 2018
Presentation Overview

- Saliency concept and motivation
- Gaussian pyramid
- Salient feature extraction
- Improvements
Saliency

- Biology: humans (aka animals) need to focus on stimuli relevant to their survival
- Saliency ~ the most interesting stimulus
  - music in subway station
  - chocolate chip cookie smell
  - painting on blank wall
- Saliency in Image processing/computer vision:
  - Portion of image most interesting / important for humans

**Saliency Map** = grayscale image where higher intensities correspond to regions of interest for humans

Original image (top) and saliency map (bottom) [2].
Algorithm Overview

1. Calculate **Gaussian pyramids** for:
   a. Intensity
   b. Color
   c. Orientation

2. Combine images from different levels of the Gaussian pyramids to create **conspicuity maps**

3. Merge conspicuity maps to create **saliency map**

4. Remove most salient region, identify other most salient region,

5. Return to step 4
**Gaussian Pyramid**

**Idea:**

1. Apply Gaussian smoothing to image A, gives image A'  
2. Create a new image A'' by downsampling A' by a factor of 2  
3. Go to step 1, starting with A''
Combine Intensity Images from Different Levels of Pyramid

- **Operations:**
  - Cross-scale subtraction/addition:
    \[
    I(c, s) = |I(c) ⊕ I(s)|
    \]
  - Normalization:
    \[
    \mathcal{N}(I) = (M - \bar{m})^2 I
    \]
    \[
    M = \text{global max of } I, \quad \bar{m} = \text{average local maxima of } I
    \]

- **Intensity Conspicuity Map:**
  \[
  \bar{I} = \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} \mathcal{N}(I(c, s))
  \]
  Upsample to allow addition and subtraction of different scale images
Cross-Scale subtraction example:

\[ \mathcal{I}(c, s) = |I(4) \odot I(1)| \]
Repeat Similar Procedure for Color Channels

- Image has RGB channels, create a Yellow channel

\[ Y = \frac{R + G}{2} - \frac{|R - G|}{2} - B \]

- Gaussian pyramid for each color channel

\[ R(\sigma), G(\sigma), B(\sigma) \text{ and } Y(\sigma) \quad \sigma \in [0..8] \]
Repeat Similar Procedure for Color Channels

- Compute “color-opponents”

\[
\mathcal{RG}(c, s) = |(R(c) - G(c)) \odot (G(s) - R(s))|
\]

\[
\mathcal{BY}(c, s) = |(B(c) - Y(c)) \odot (Y(s) - B(s))|
\]

- Compute conspicuity map for “color-opponent” channels

\[
\overline{C} = \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} \left[ N(\mathcal{RG}(c, s)) + N(\mathcal{BY}(c, s)) \right]
\]

**ALGORITHM SUMMARY**

1) Difference of color channels at scale \( c \)
2) Difference of color channels at scale \( s \)
3) Upsample to perform subtraction between different scales
4) Normalize
5) Sum to create color conspicuity map
Repeat Similar Procedure for Orientation

- Compute Gabor Pyramid
  - Gaussian pyramid with additional complex exponential term in low pass filter [4]
    \[
    \mathcal{O}(\sigma, \theta) = \text{Gaussian LPF} \left( e^{i\theta(x+y)} \cdot I(x, y) \right)
    \]
    \[
    \theta = \{0, 45, 90, 135\}, \quad \sigma = [0, 1, \ldots, 8]
    \]

- Combine different layers from pyramid to create conspicuity map
  \[
  \mathcal{O}(c, s, \theta) = |\mathcal{O}(c, \theta) \ominus \mathcal{O}(s, \theta)|
  \]
  \[
  \mathcal{\overline{O}} = \sum_{\theta \in \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}} \mathcal{N} \left( \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} \mathcal{N}(\mathcal{O}(c, s, \theta)) \right)
  \]
Saliency Map

- Normalize and add the conspicuity maps to get the saliency map:

\[
S = \frac{1}{3} \left( \mathcal{N}(\bar{I}) + \mathcal{N}(\bar{C}) + \mathcal{N}(\bar{O}) \right)
\]

![Saliency Map Diagram](image-url)
Algorithm Recap:

1. Calculate **Gaussian pyramids** for:
   a. Intensity
   b. Color
   c. Orientation

2. Combine images from different levels of the Gaussian pyramids to create **conspicuity maps**

3. Merge conspicuity maps to create **saliency map**

4. Remove most salient region, identify other most salient region,

5. Return to step 4
Algorithm Example:

Input image

Linear filtering

- Colors
- Intensity
- Orientations

Center-surround differences and normalization

- Feature maps (12 maps)
- Intensity maps (6 maps)
- Orientation maps (24 maps)

Across-scale combinations and normalization

- Conspicuity maps

Linear combinations

Saliency map

Winner-take-all

Inhibition of return

Attended location

Input image

Output (FOA)

92 ms

145 ms
Example: noise tolerance
Improvement: **Variable weighting** for Intensity, Orientation, Color

Three Components:

- **Salient point set determined by a threshold**
- **Spatial compactness**
  - Require salient region of conspicuity map to be enclosable in polygon
  - Smaller area polygon -> more salient
- **Saliency density**
  - Want high variability in salient regions

Example: Only color map used in saliency map
**Improvement:** *Variable weighting* for Intensity, Orientation, Color

Better defined salient regions are obtained by variable weighting of color, intensity, and orientation maps
Direct comparison: **Variable vs. Constant** weighting of feature maps

**Fig. 4.** (a) Original Image; (b) Saliency Map using Itti et al’s model [4]; (c) Cropped Image using Itti et al’s model [4]; (d) Saliency Map using CSI; (e) Cropped Image using CSI.
References

Bonus: Saliency-Aware Autonomous Exploration… here at UNR

https://www.youtube.com/watch?v=-ReBwdzJoIM

- Autonomous Robots Lab (Dr. Alexis)
- This work done by PhD. Candidate Tung Dang