Saliency-driven Omnidirectional Imaging adaptive Coding : Modeling

A summary by Bhishma Dedhia

May 22, 2018

1 Introduction

This paper proposes a saliency-driven omnidirectional imaging adaptive coding solution where a saliency detection model drives the quantization process.The idea is to invest more quality on the regions where the viewer tends tofixate his/her attention and vice-versa, thus reducing the overall bitrate for atarget perceptual quality. The coding performance of the proposed solutionis assessed by a novel objective quality metric for omnidirectional imaging,which is validated by formal subjective testing

2 Architecture of the Saliency detection Model

Figure 1 shows the model for Saliency detection. The model has been based on images from the Salient360 Grand challenge. The dataset consists of 20 omnidirectional images. It consists of 3 branches:

- Ground truth branch
- Latitude driven SDM: VLIM Generation
- Proposed SDM: LVSM Generation

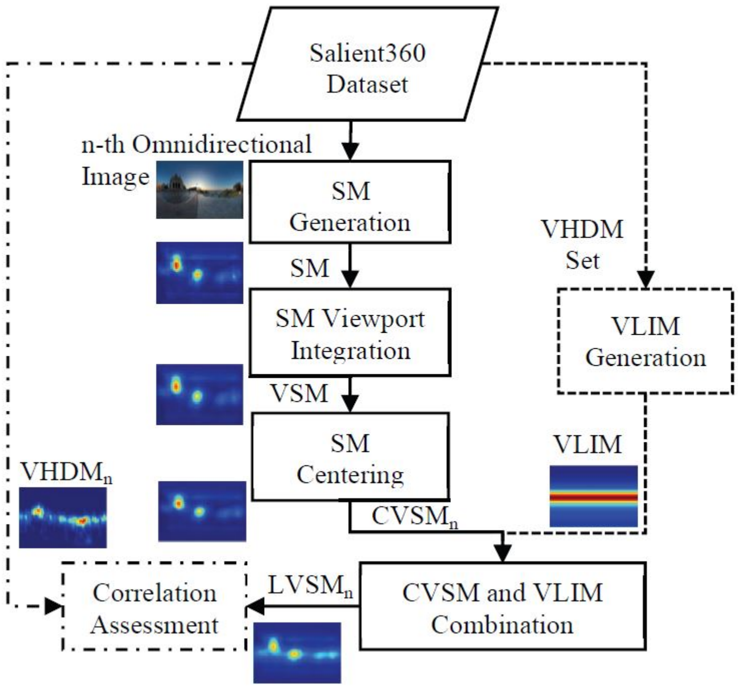

Figure 1: Saliency Detection Architecture

2.1 Ground truth branch

The ground truth branch on the left represents the saliency ground truth for each omnidirectional image, in this case expressed by means of the so-called Viewport integrated Head Direction Map (VHDM) which has been experimentally collected for the Salient360! Grand Challenge.

2.2 Viewport based Latitude Importance Map (VLIM)

The Salient360! challenge allows to extract some globally true statistics for the images based on latitude. This is independent of the images and has been experimentlly collected based on viewer attention.

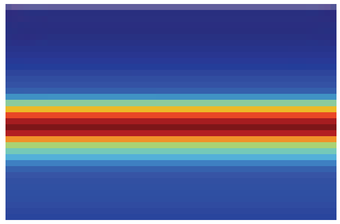

Figure 2: VLIM

2.3 Proposed SDM : LVSM Generation

2.3.1 Saliency Map generation

This is done using MLNET: a CNN trained to generate Saliency maps for 2D images. This is the first step towards generating the LVSM. Since the model has been trained for 2D images further processing is required to adapt it for omnidirectional images.

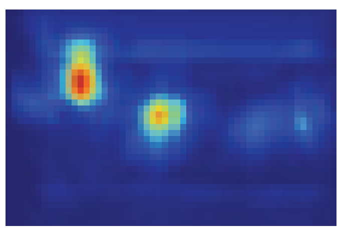

Figure 3: Saliency Map

2.3.2 Saliency Map viewport integration

The user's head and eye direction are not always same. The eyes can move around even if the head is fixed. This effect is accounted for by calculating the saliency of a point based on the saliency of its neighbours.

Each pixel in the ERP is converted to its spherical co-ordinates and this direction is considered as the centre of the viewport. The saiency of this pixel is calculated using the weighted average of all the pixels in the viewport area. A 2D gaussian function with stan. deviation of 4 degree is used to calculate the weights. This is done for all the pixels to finally obtain the Viewport integrated Saliency Map. (VSM)

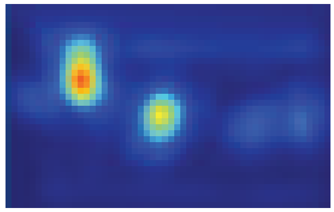

Figure 4: VSM

2.3.3 Saliency Map centering

This module takes into account that viewers don't move their head to fixate regions near the poles. Hence this module moves previous VSM generated to Latitudes closer to the equator. Thus we finally obtain the Centered Viewport based Saliency Map (CVSM).

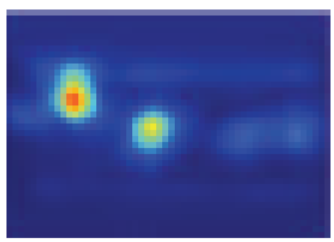

Figure 5: CVSM

2.3.4 CVSM and VLIM Combination

The regions near the equator are generally more viewed than the ones away from it. This bias is introduced using the VLIM.

$$LVSM = w \times CVSM + (1 - w) \times VLIM$$

$$w = 0.8$$

2.3.5 SDM performance assessment

The proposed SDM was compared to the VHDM using Root mean square error. The results obtained were:

	VLIM	VSM	CVSM	LVSM
Average RMSE	0.3298	0.1638	0.1538	0.1348

Table 1: SDM results

3 Saliency based adaptive Quantization Model

An adaptive QP based on Saliency and spatial activity may help in the better allocation of bitrate and improve the efficiency of coding. Smaller QP is assigned to highly salient regions. Similarly a high QP should not be assigned to quantization sensitive regions (like uniform or slowly varying areas). Hence the QP for the i^{th} LCUis given by:

$$QP_i = round(\frac{QP_{slice}}{\sqrt{w_i}})$$

Here QP_{slice} is the default QP defined for the current slice . The w_i is given by:

$$w_i = \begin{cases} a + \dfrac{b}{1 + \exp\left(-c \times (S(X_i)/n - \bar{S})/\bar{S}\right.} & ,if \ l \leq 10 \\ a + \dfrac{b}{1 + \exp\left(-c \times (S(X_i) - \bar{S})/\bar{S}\right.} & ,if \ l > 10 \end{cases}$$

$$a = 0.7, b = 0.6, c = 4$$

$\bar{S}$: Average saliency for the whole set of LCU within the frame
$S(X_i)$: Average Saliency for the i-th LCU
The normalized spatial activity n is calculated using:

$$n = \frac{h \times l + t}{l + h \times t}$$

$$h = 2^{\frac{r}{6}}, r = 6$$

l is the spatial activity in a Luma CB and helps us identify regions which are sensitive to quantization noise. l is given by :

$$l = 1 + min(\sigma^2_{Y,k}), k = 1, .., 4$$

Here $\sigma_{Y,k}$ is the spatial activity in sub block k of the Luminance CB. For this reason, for low spatial activity l, the normalized spatial activity is considered to compute w_i and reduce the QP in these regions.