Dynamic Visual Attention on the Sphere

Iva Bogdanova[a] *, Alexandre Bur[a], Heinz Hügli[b] and Pierre-André Farine[a]. †

[a]Ecole Polytechnique Fédérale de Lausanne (EPFL),
Institute of Microengineering (IMT),
Pattern Recognition Laboratory (PARLAB)
Rue A.-L. Breguet 2, 2000 Neuchâtel, Switzerland

[b]University of Neuchâtel (UniNE),
Switzerland

In this paper we present a computational model of dynamic visual attention on the sphere which combines static (intensity,chromaticity, orientation) and motion features in order to detect salient locations in omnidirectional image sequences while working directly in spherical coordinates. We build the motion pyramid on the sphere by applying block matching and varying the block size. The spherical motion conspicuity map is obtained by fusing together the spherical motion magnitude and phase conspicuities. Furthermore, we combine this map with the static spherical saliency map in order to obtain the dynamic saliency map on the sphere. Detection of the spots of attention based on the dynamic saliency map on the sphere is applied on a sequence of real spherical images. The effect of using only the spherical motion magnitude or phase for defining the spots of attention on the sphere is examined as well. Finally, we test the spherical versus Euclidean spots detection on the omnidirectional image sequence.

1. INTRODUCTION

Visual attention (VA) is the ability of the human visual system (HVS) to rapidly select the most salient objects in a given scene. VA represents also a fundamental mechanism for computer vision where similar speed up of the processing can be envisaged. The motion is clearly involved in visual attention, where rapid detection of moving objects is essential for adequate interaction with the environment [1]. Over the last decade, several investigations focused on the architecture of the computer model of dynamic visual attention. In order to deal with image sequences, the current dynamic models generally integrate additional motion components to the classical saliency-based model proposed in [2]. A dynamic model that combines the static features (intensity, color, orientation) and dynamic features (temporal changes) for four orientations $(0^0, 45^0, 90^0, 135^0)$ is considered in [3]. There a comparison with the human vision is performed experimentally, by comparing the models

*IB is currently with Embedded Systems Laboratory (ESL) at EPFL.

†The authors acknowledge the support of the Swiss National Center for Competence in Research IM2 and Swiss National Science Foundation under contract FN-108060.

with respect to the eye movement patterns of human subjects. Combining in different ways color, intensity, orientation and motion magnitude is examined in [4] and an evaluation is performed on single synthetic sequences. Another dynamic model based on motion contrast that is computed as the difference between local (hierarchical block-matching) and dominant motion is proposed in [6]. The main conclusion withdrawn from these works is that the motion contrast is much more relevant than other features for predicting human attentional behavior.

A feature contrast in the sense of VA refers to a contrast between a center and surrounding region according to a specific feature such as intensity, chromaticity, orientation. Consequently, the motion contrast refers to the difference in the motion between a center and surrounding region. It was illustrated in [7] that the attention is linked to the motion contrast in magnitude, since each direction is represented in a hierarchical way by several speed selectivities ranges. In addition, this representation is composed of a set of direction selectivities, which are linked to motion contrast in phase.

There are several ways for computing the motion conspicuity maps but all of them serve conventional images, i.e. images obtained with conventional cameras and which we call Euclidean. By now, there is no proposed algorithm for computing dynamic visual attention in omnidirectional video. Nowadays, the demand of omnidirectional imaging is increasing because of its larger field of view and it is widely used in robotics, surveillance etc. It is clear that the omnidirectional sensors are related to the spherical geometry. In fact, there exists an equivalence between the central catadioptric projecttion and the two-step mapping onto the sphere [8]. On the other hand, the multi-camera sensors output images directly in spherical coordinates [9].

This paper presents a computational model of dynamic visual attention on the sphere which combines static and dynamic features in order to detect salient locations in omnidirectional image sequences. More specifically, it extends our previous work by providing a methodology for computing motion contrast and motion conspicuity that is suited to image sequences in spherical coordinates. Consequently, this model computes a spherical saliency map that is related to static features and a saliency map derived from motion scene features and then combines them into a dynamic saliency map which encodes stimulus saliency. The different steps of the model are shown on Figure 1 and explained in the following sections. Furthermore, the effect of the motion magnitude and phase is tested with respect to the final spherical saliency map. For this reason, spherical video with still and moving background is used.

The paper is organized as follows. In section 2, we remind the basic steps in the static VA on the sphere computed using the intensity, chromatic and orientation features. Then, we present an algorithm for computing the motion saliency map on the sphere in section 3. Finally, the fusion of the static and motion spherical saliency maps and obtaining the dynamic saliency on the sphere is presented in 4. We apply the proposed algorithm on a real omnidirectional image sequence obtained by a multicamera system in Section 5 and we test the spherical versus Euclidean spots detection on the same sequence based on our algorithm and on the Euclidean VA, respectively.

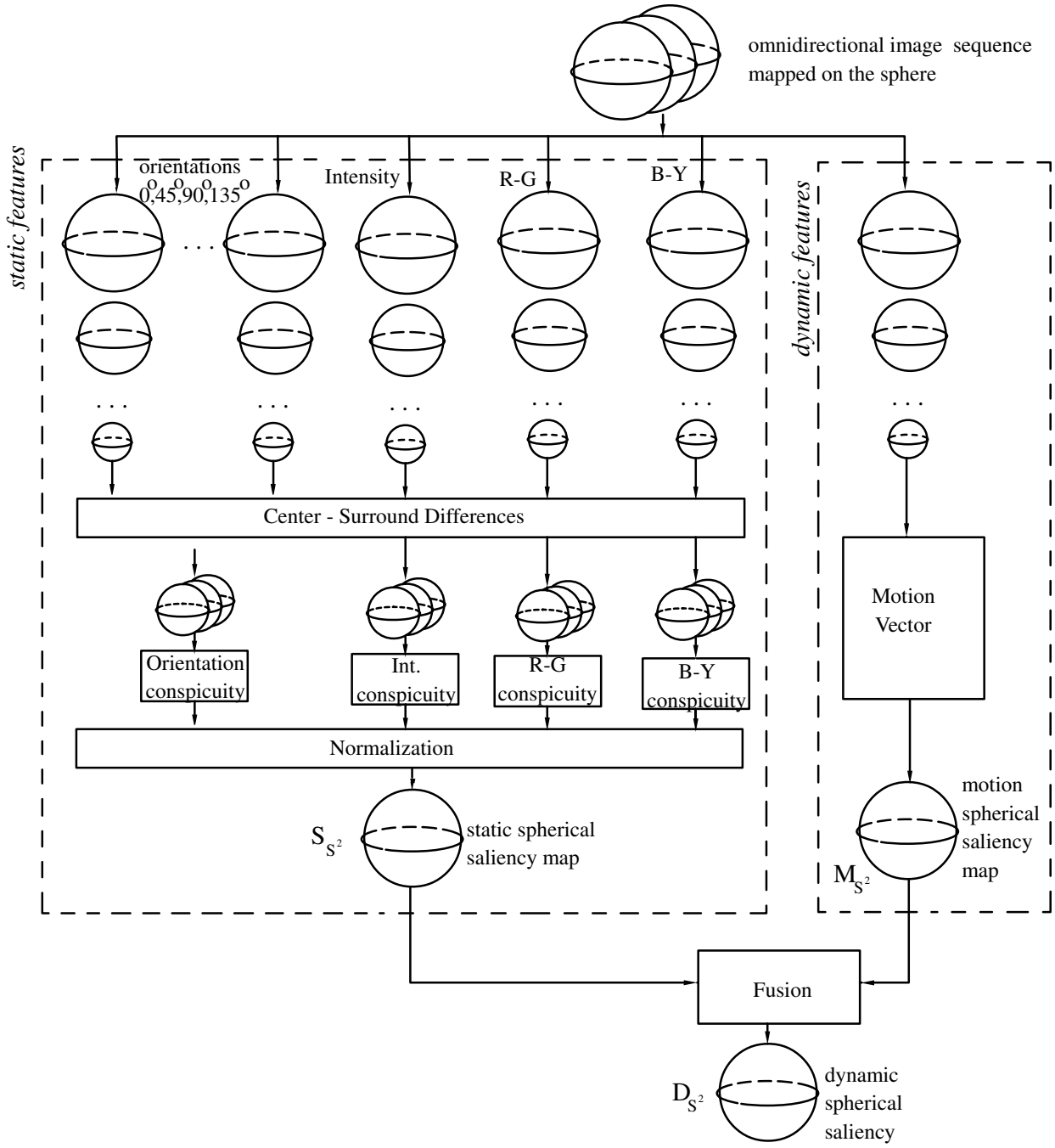


Figure 1. Model of dynamic VA on the sphere which combines static and motion scene features in order to detect salient locations in omnidirectional image sequences.

2. STATIC SPHERICAL SALIENCY MAP

Now, let us define the procedures for obtaining the different cue conspicuity maps on the sphere which are then combined in order to obtain the static spherical saliency map.

Let us have as an input a spherical image $f(\theta, \varphi) \in L^2(S^2)$ with $\theta \in [0, \pi]$ and $\varphi \in [0, 2\pi)$ and of bandwidth β such that its Fourier coefficients $\hat{f}(l, m) = 0, \forall l > \beta$. In fact, this image is defined on $2\beta \times 2\beta (\beta \in \mathbb{N})$ squared grid of, respectively, equi-angular resolution in θ and φ:

$$\mathcal{G} = \left\{ (\theta_p, \varphi_q) : \theta_p = \frac{(2p+1)\pi}{4\beta} \;\; \varphi_q = \frac{q\pi}{\beta}, p, q \in \mathbb{Z}[2\beta] \right\}. \tag{1}$$

Furthermore, each of the spherical image features are extracted. Then, a conspicuity map for each feature is built. Finally, the spherical saliency map is obtained by fusing together all the spherical cue conspicuity maps. Precise details on computing the static spherical saliency map can be found in [10] but here we remind some basic points.

2.1. Computing Several Features on the Sphere

First, we need to define each of the static features $j = 1 \cdots 7$ in the spherical image, f_j as follows:

1. intensity: $f_{int} = 0.3r + 0.59g + 0.11b$;
2. yellow-blue: $f_{BY} = \frac{(b - r/2 - b/2)}{f_{int}}$;
3. red-green: $f_{RG} = \frac{(r-g)}{f_{int}}$;
4. four orientations: $f_{0^0}, f_{45^0}, f_{90^0}, f_{135^0}$. They are obtained applying a Gabor pyramid on f_{int} where the spherical Gabor filter [5] reads

$$\psi_{gabor}(\theta, \varphi, \phi) = \lambda^{1/2}(\theta, a) e^{ik_0 \frac{2}{a} \tan\frac{\theta}{2}} \cos(\phi - \varphi) e^{-\frac{2}{a^2}\tan^2\frac{\theta}{2}} \big(1 + \frac{1}{a^2}\tan^2\frac{\theta}{2}\big), \tag{2}$$

where $\theta \in [0, \pi]$, $\varphi \in [0, 2\pi)$, $\phi \in \{0^0, 45^0, 90^0, 135^0\}$, respectively and $k_0 = 30, a = 0.03$ and $\lambda(\theta, a)$ is a normalization factor;

2.2. Spherical Conspicuity Map for Each Feature

Let us have a spherical image f_0 defined on a grid of size $2^{n+2} \times 2^{n+2}$. The procedure for computing the spherical feature conspicuity map C_j relies on the spherical Gaussian pyramid and the center-surround mechanism. It includes the following steps:

1. Construct the n-level spherical gaussian pyramid ;
2. Compute the multiscale maps :

$$\begin{aligned} \mathcal{M}_1 &= |f_2 \ominus f_5|, \mathcal{M}_2 = |f_2 \ominus f_6|, \mathcal{M}_3 = |f_3 \ominus f_6|, \\ \mathcal{M}_4 &= |f_3 \ominus f_7|, \mathcal{M}_5 = |f_4 \ominus f_7|, \mathcal{M}_6 = |f_4 \ominus f_8|, \end{aligned} \tag{3}$$

where $\ominus$ refers to a cross-scale difference operator that interpolates the coarser scale to the finer one and then performs a point-by-point substraction;

3. Compute the weight coefficients w_{S^2} and normalize the maps as follows

$$\mathcal{N}_{S^2}(C(\theta,\varphi)) = w_{S^2} \cdot C(\theta,\varphi), \tag{4}$$

with

$$w_{S^2} = \frac{4\pi Max(C(\theta,\varphi))}{\sum_\theta \sum_\varphi C(\theta,\varphi)\sin\theta},$$

where $C(\theta,\varphi)$ is the corresponding spherical conspicuity map;

4. Compute the final *spherical feature conspicuity map* C_j using

$$C_j = \sum_{k=1}^{n-3} \mathcal{N}(\mathcal{M}_k), \tag{5}$$

where $\mathcal{M}_k$ is the multiscale map defined above.

This procedure is applied to each of the features in order to compute seven spherical conspicuity maps $C_j, j = 1 \cdots 7$.

2.3. Spherical Cue Conspicuity Maps

Using all feature conspicuity maps as obtained in Section 2.2, the following cue conspicuity maps are computed:

1. $C_{int} = C_1$;

2. $C_{chrom} = \frac{\mathcal{N}_{S^2}(C_2)+\mathcal{N}_{S^2}(C_3)}{2}$, where C_2 is the red-green spherical conspicuity map and C_3 is the yellow-blue conspicuity map. They are normalized according Equation (4) ;

3. $C_{orient} = \frac{\mathcal{N}_{S^2}(C_4)+\mathcal{N}_{S^2}(C_5)+\mathcal{N}_{S^2}(C_6)+\mathcal{N}_{S^2}(C_7)}{4}$, where C_4, C_5, C_6, C_7 are obtained after applying the procedure in Section 2.2 on the four orientation features from Section 2.1-4, and normalized according Equation (4).

Finally, the static spherical saliency map is computed by fusing together all cue conspicuity maps obtained in Section 2.3:

$$S_{S^2} = \sum_{cue \in int,chrom,orient} \mathcal{N}(C_{cue}), \tag{6}$$

where $\mathcal{N}()$ is the normalization step according to Equation (4). Due to the different nature of the spherical cue conspicuity maps, the conspicuity cues are previously scaled at the same range values by applying a peak-to-peak normalization.

3. MOTION SPHERICAL SALIENCY MAP

Since motion is involved in the VA mechanism, a computational VA model that is designed for omnidirectional video sequences must consider both static and dynamic features on the sphere. The dynamic spherical saliency map D_{S^2} is meant to highlight the moving scene constituents on the sphere, and consequently in omnidirectional images. In fact, we assume that the location where something moves is salient.

From a physical point of view, the motion $\vec{v}$ at a location on the sphere $\omega \equiv (\theta, \varphi) \in S^2, \theta \in [o, \pi], \varphi \in [0, 2\pi)$ and time t is defined as the derivative of ω over the time, which reads

$$\vec{v}(\omega, t) = \lim_{\Delta t \to 0} \frac{\vec{d}}{\Delta t} = \frac{d\omega}{dt}, \tag{7}$$

i.e. the motion can be seen as the displacement $\vec{d}$ on the sphere by an infinitesimal time interval Δt.

From image processing point of view, each image pixel corresponds to the intensity value obtained by projection of the 3D space onto the image plane. Motion can be induced either by the displacement of the object in 3D space or the displacement of the sensor plane, or both. Due to the discrete nature of the video stream, motion $\vec{v}$ is defined as the displacement between two consecutive frames (spherical images):

$$\vec{v} = \frac{d}{\Delta t}, \tag{8}$$

and the motion in the omnidirectional image plane is often represented by a vector field $\mathbf{M} = \{\vec{v}(\omega)\}$.

We compute the motion vector at different scales taking advantage of the spherical gaussian pyramid. This multi-scale approach allows the detection of small displacements at fine scales, whereas large displacements are detected at coarser scales.

In general, the relative motion is the difference between the local and the dominant motion. The local motion $\vec{V}_{local}$ at each point on the sphere $\omega \equiv (\theta, \varphi)$ (or the motion vector) can be obtained by, for instance, a block matching. It is computed through series of levels (resolutions) each providing the input for the next. Moreover, on each resolution, the block matching is done for a certain neighborhood (window) size, that increases with the hierarchy level. In fact, the local motion does not necessary represent the motion contrast but it is the case when the dominant motion is null, i.e. the camera is fixed.

3.1. Block Matching on the Sphere

An approach for computing the motion estimation in spherical images integrated into a multi-resolution scheme was first developed in [11] in order to find a prediction image. Here we concentrate on simple block- matching, i.e. it takes place at a single level and we don't need to propagate the motion vector in a pyramidal structure. It is clear that the local motion estimation algorithm takes place in the spherical domain. The algorithm simply pairs solid angles from two spherical signals and we call it spherical block matching algorithm (sBMA). It aims at computing the motion field between two spherical images.

Let us define two spherical images as $f(\theta, \varphi)$ and $g(\theta, \varphi)$ and sample them on an equiangular grid:

$$\mathcal{G}_j = \{(\theta_{jp}, \varphi_{jq}) \in S^2 : \theta_{jp} = \frac{(2p+1)\pi}{4B_j}, \varphi_{jq} = \frac{q\pi}{B_j}\}, \tag{9}$$

$p, q \in \{n \in \mathbb{N} : n < 2B_j\}$ for some range of bandwidth $B = \{B_j \in 2\mathbb{N}, j \in \mathbb{Z}\}$. This grid allows us to perfectly sample any band-limited function $f \in L^2(S^2)$ of bandwidth B_j. Moreover, this class of sampling grids is associated to a fast spherical fourier transform.

Algorithm 1 Block matching on the sphere

$\mathbf{M}_i = [0,0], \forall i, \delta_\theta = \frac{\pi}{2B}, \delta_\varphi = \frac{2\pi}{2B}, B \equiv$ full resolution
divide g into I uniform blocks of size $M\delta_\theta \times N\delta_\varphi$;
$i = 0$;
repeat
  $(p_i, q_i) \leftarrow$ position of g_i;
  $\Omega \leftarrow \{(p,q)\}$such that
  $p \in [p_i + \mathbf{M}_i(1) - \frac{W\delta_\theta}{2} + 1, p_i + \mathbf{M}_i(1) + \frac{W\delta_\theta}{2}]$ and
  $q \in [q_i + \mathbf{M}_i(2) - \frac{W\delta_\varphi}{2} + 1, q_i + \mathbf{M}_i(2) + \frac{W\delta_\varphi}{2}]$;
  $f_i = \arg\min_\Omega MSE(g_i, f_i)$;
  $(w_i, t_i) \leftarrow$ position of f_i;
  $\mathbf{M}_i \leftarrow [p_i + w_i, q_i + t_i]$;
  $i \leftarrow i + 1$;
until $i > I$

Then, the spherical image g is divided into uniform solid angles of size $M\delta_\theta \times N\delta_\varphi$ that form blocks g_i. These are paired with the best matching blocks with the same size in the reference spherical image f within a search window of size $W\delta_\theta \times W\delta_\varphi$ around the location of the block g^i. A full search of each block g_i determines the corresponding motion vectors $\mathbf{M}$. We must note that even the blocks g_i are all distinct, the blocks f_i may be overlapping. This algorithm also takes into account the periodicity in azimuthal direction.

3.2. Motion Pyramid on the Sphere

In the sense of VA, center-surround contrast refers to a difference between a center and surround region. Regarding static features, the alternative approach proposed in [10] approximates the multiscale center-surround contrast using pyramid and cross-scale differences. What concerns the motion feature, a parallel approach can be performed similarly. The basics of the center surround contrast computation lay in creating a pyramidal structure on the motion. In fact, computing the motion field at different levels of this pyramid on the sphere is obtained by varying the block size in the sBMA.

In order to compute motion contrasts, the idea is basically to define two average motion vectors from the motion pyramid, $\vec{v}_c$ and $\vec{v}_s$, which represent the motion of center and surround regions, respectively. The multiscale motion field pyramid $\mathbf{M}_n$ is composed of

n motion maps $\mathbf{M}_i$ corresponding to the motion estimation at diferent scales $i = 1 \cdots n$. Coarse scale maps detect motion of large regions while fine scale maps detect motion of small regions. The initial resolution of the first level $\mathbf{M}_1$ is $m_1 \times n_1$ and the resolution of the consecutive levels is decreasing over the pyramid by a factor two between each level. Average motion vectors $\vec{v}_c$ and $\vec{v}_s$ are obtained from $\mathbf{M}_n$ according to their corresponding levels.

3.3. Motion Conspicuity Operators

Once the motion average vectors $\vec{v}_c$ and $\vec{v}_s$ have been estimated from the motion pyramid, the next step toward computing the motion conspicuity is to apply on them a motion conspicuity operator in order to detect center-surround contrasts. There are three possible such operators:

3.3.1. Motion contrast in phase and magnitude

This operator computes the norm of the motion vector difference:

$$D_{cs}(\vec{v}_c, \vec{v}_s) = \|\vec{v}_c(\theta, \varphi)\| - \|\vec{v}_s(\theta, \varphi)\|, \tag{10}$$

It is clear that it considers both the phase and the magnitude of the motion vector.

3.3.2. Motion contrast in magnitude

It consists in computing the norm of the center and surround motion vectors and then to take the absolute difference. This reads:

$$A_{cs}(\vec{v}_c, \vec{v}_s) = |\|\vec{v}_c(\theta, \varphi)\| - \|\vec{v}_s(\theta, \varphi)\||, \tag{11}$$

where $\vec{v}_c$ is the motion vector at the center level M_i and $\vec{v}_s$ is the motion vector at the surround level which is up-sampled to the corresponding resolution.

3.3.3. Motion contrast in phase

This operator is sensitive to motion phase difference.

$$B_{cs}(\vec{v}_c, \vec{v}_s) = \mathcal{O}\Big(\arctan \frac{v_{c\varphi}}{v_{s\theta}} - \arctan \frac{v_{c\varphi}}{v_{s\theta}} \Big), \tag{12}$$

$$\mathcal{O}(\alpha) = \begin{cases} \alpha, \text{ if } 0 \leqslant \alpha \leqslant \pi \\ (2\pi - \alpha), \text{ otherwise,} \end{cases} \tag{13}$$

where $(v_{c\theta}, v_{s\varphi})$ and $(v_{c\theta}, v_{s\varphi})$ are the θ, φ components of the vectors $\vec{v}_c$ and $\vec{v}_s$. $\mathcal{O}(\alpha)$ is used to shift the phase difference when the phase exceeds π.

3.4. Algorithm for Computing the Spherical Motion Conspicuity Map

In this section we propose an algorithm for computing the spherical motion conspicuity map using spherical motion pyramid. Its block diagram is shown on Figure 2 and comprises the following steps.

- Build the multi-scale motion pyramid on the sphere $\mathbf{M}$, as in section 3.2;

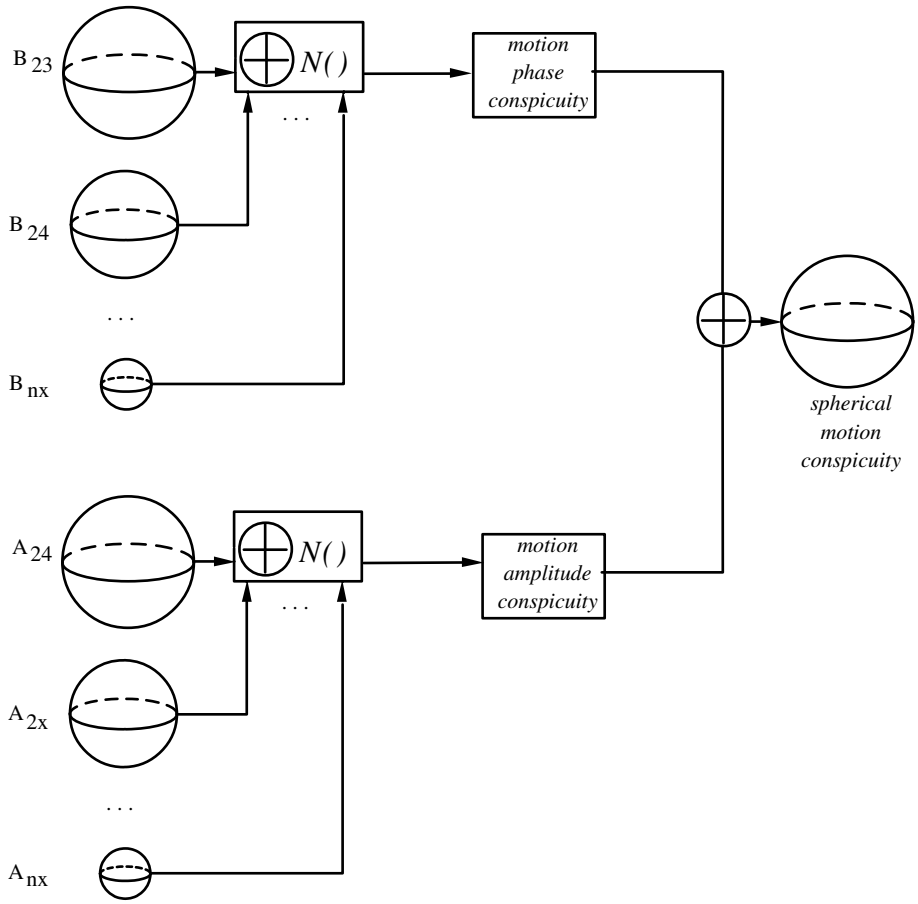


Figure 2. The spherical phase and magnitude motion model based on the motion pyramid on the sphere. The spherical motion conspicuity map is computed with decoupled phase and magnitude contrast.

- Compute the spherical motion magnitude conspicuity map $C_{S^2{}_{magitude}}$ by applying the magnitude conspicuity operator A_{cs} from Equation 11 to the multi-scale motion pyramid $\mathbf{M}_i$;
- Compute the spherical motion phase conspicuity map $C_{S^2{}_{phase}}$ by applying the phase conspicuity operator B_{cs} from Equation 12 to the multi-scale motion pyramid $\mathbf{M}_i$;
- Combine in classical manner the two previously obtained conspicuity maps and obtain the motion spherical conspicuity map:

$$M_{S^2{}_{p\&m}} = \mathcal{N}(C_{S^2{}_{phase}}) + \mathcal{N}(C_{S^2{}_{magitude}}), \tag{14}$$

where $C_{S^2{}_{phase}} = \sum_{ij} \mathcal{N}(B_{ij})$ and $C_{S^2{}_{magnitude}} = \sum_{ij} \mathcal{N}(A_{ij})$.

The intermediate phase conspicuity maps B_{ij} highlight phase differences between a center level i and a surround level j, while the intermediate magnitude conspicuity maps A_{ij} highlight the magnitude differences at the same corresponding levels. Furthermore, the obtained motion map on the sphere is combined with the static spherical saliency map in order to obtain the dynamic saliency map on the sphere as explained in the next section.

4. FUSION OF STATIC AND MOTION SPHERICAL MAPS

In this section, we aim at integrating the static spherical saliency map S_{S^2} and the dynamic one M_{S^2} in order to obtain the final spherical saliency map D_{S^2}. Such a combination is difficult but necessary when several maps are considered. This yields a single measure of interest for each location regardless of which feature has contributed to the saliency.

Indeed, the static saliency map is computed on each sphere from the sequence, while the dynamic one is computed on two successive spheres. The data driven competition mechanism presented in Section 2 is a suitable integration concept for both cues. Therefore, the final spherical saliency map is computed according the equation

$$D_{S^2} = w_s S_{S^2} + w_d M_{S^2}, \tag{15}$$

where the coefficients w_s ans w_d are combuted according weighting function w presented in (4).

Once we have obtained the final spherical saliency map, the most salient locations on the sphere are selected by applying the "Winer-Take-All" (WTA) network in the same manner as in the case of static visual attention on the sphere. The complete details can be found in [10].

5. EXPERIMENTAL RESULTS

In this section we first apply the spherical dynamic visual attention algorithm on real spherical videos. We have tested it on ten different sequences but here we show only some of them. The working sequences are obtained with multi-camera omnidirectional sensor [9], which directly outputs them in spherical coordinates (θ, φ). Each of these images is

defined on 1024×1024-spherical grid and cover almost 75% from the sphere. The first experiment is straightforward and concerns a general application of our algorithm to real omnidirectional image sequences. Second, we examine the effect of the amplitude and the direction (phase) of the spherical motion on computing the spherical saliency map and consequently their affect on defining the spots of attention. Finally, we compare our algorithm to the standard one by applying the spherical and conventional dynamic visual attention algorithm on the same omnidirectional video.

5.1. Spots of attention in Real Omnidirectional Video

Let us take a sequence of 17 spherical images. The scene is taken in an office where the camera is pointing down the table on which a toy-car is moving. While crossing the table, the car passes through the north pole of the sphere. For this example, we show some of the intermediate results for better illustration of the entire dynamic visual attention mechanism on the sphere. Applying the sBMA with a varying block size of $4 \times 4, 8 \times 8, 16 \times 16, 32 \times 32, 64 \times 64$, we obtain a five-level motion pyramid on the sphere. After applying the procedure described in Section 3.4, where both the magnitude and phase of the spherical motion field are considered, we obtain the spherical motion cue. It is shown on Figure 3 for some of the frames in this sequence.

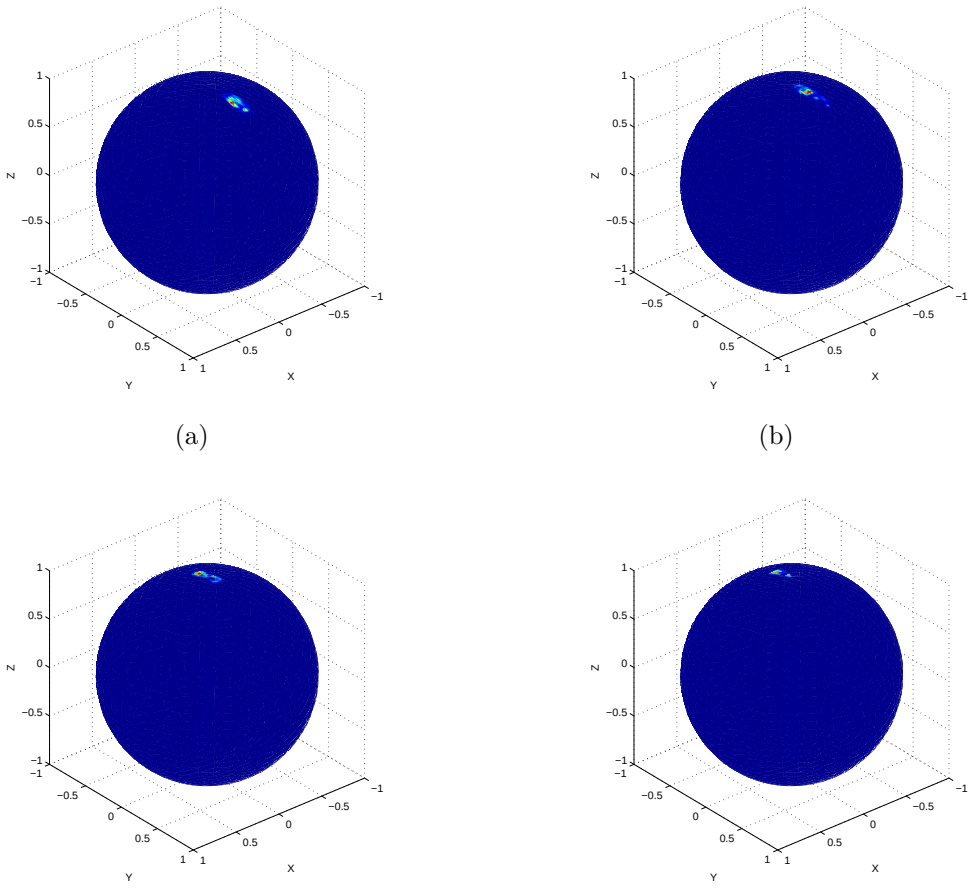
The final spherical saliency mapis obtained by summing the spherical motion and static cues and is shown on Figure 4.

Consequently, based on the final spherical saliency map, the spots of attention on the sphere are detected. The first three most salient locations on the sphere are shown on Figure 5. The spot of attention with rank one is shown in red. which in our case is the car. Again, it is important to note that even when the car is passing through the north Pole of the sphere, as expected, it is detected as salient. Furthermore, in frame 16, the car is not detected as the most salient object in the scene. In fact it is not moving anymore. In this frame, the most salient location is one of the red chairs around the table. This suggests that the motion is dominant in determining the saliency even in spherical coordinates. When there is no motion in a given scene, the saliency is determined by the color opponency.

5.1.1. Motion Magnitude versus Motion Direction on the Sphere

In order to better examine the influence of the motion magnitude and direction (phase) on the dynamic visual attention in spherical coordinates, we perform the following experiment. For building the motion cue only the motion amplitude or only the motion phase is considered. Furthermore, both ways of computing the motion cue on the sphere is applied on omnidirectional video, obtained with static and moving omnidirectional sensor, respectively, i.e. with still or moving background.

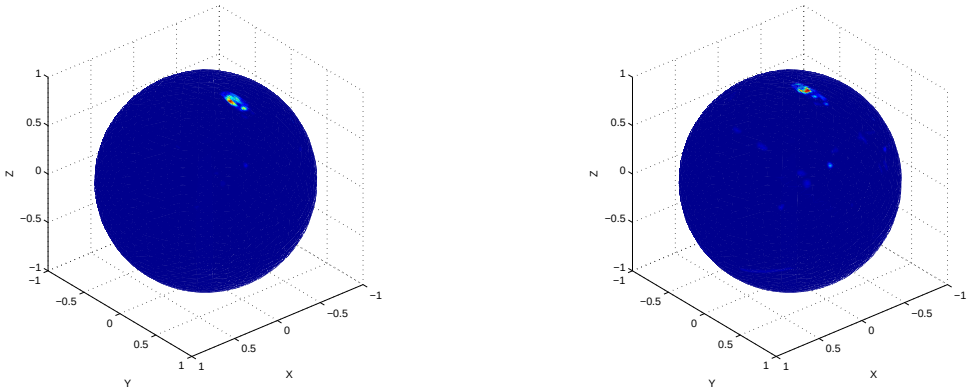
First, the still background video is examined. The motion cue is computing while considering only the spherical motion contrast in magnitude. In Figure 6 are illustrated some of the frames in the "corridor" sequence. Apart the salient objects in the posters fixed on the walls, another two simultaneous motions are available: two persons are going out of the office. For purpose of visualization the unwrapped spherical image is shown and only the first five detected spots of attention. Their rank is as follows: 1- red, 2-green, 3-blue , 4- yellow, 5 - magenda. One can easily observe that when considering only the magnitude in still background omnidirectional video, the most salient location is determined mainly



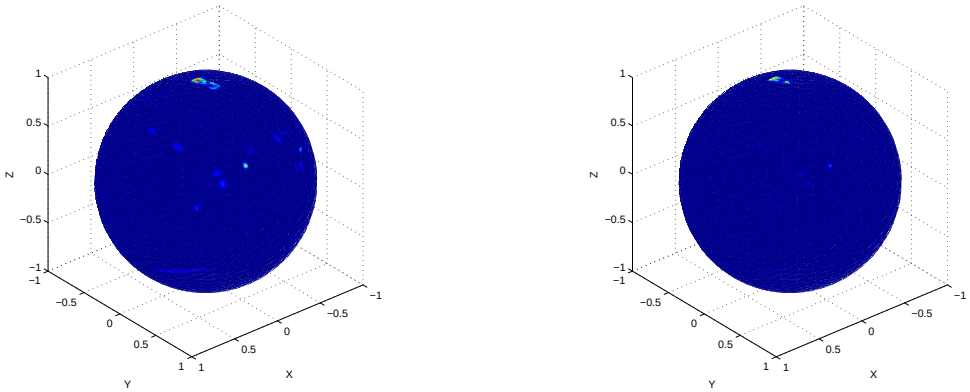
(a) (b)

(c) (d)

Figure 3. Spherical motion cue: (a) frame 8; (b) frame 10; (c) frame 12; (d) frame 14;



(a) (b)



(c) (d)

Figure 4. Final spherical saliency map on some of the frames in the sequence: (a) frame 8; (b) frame 10; (c) frame 12; (d) frame 14;

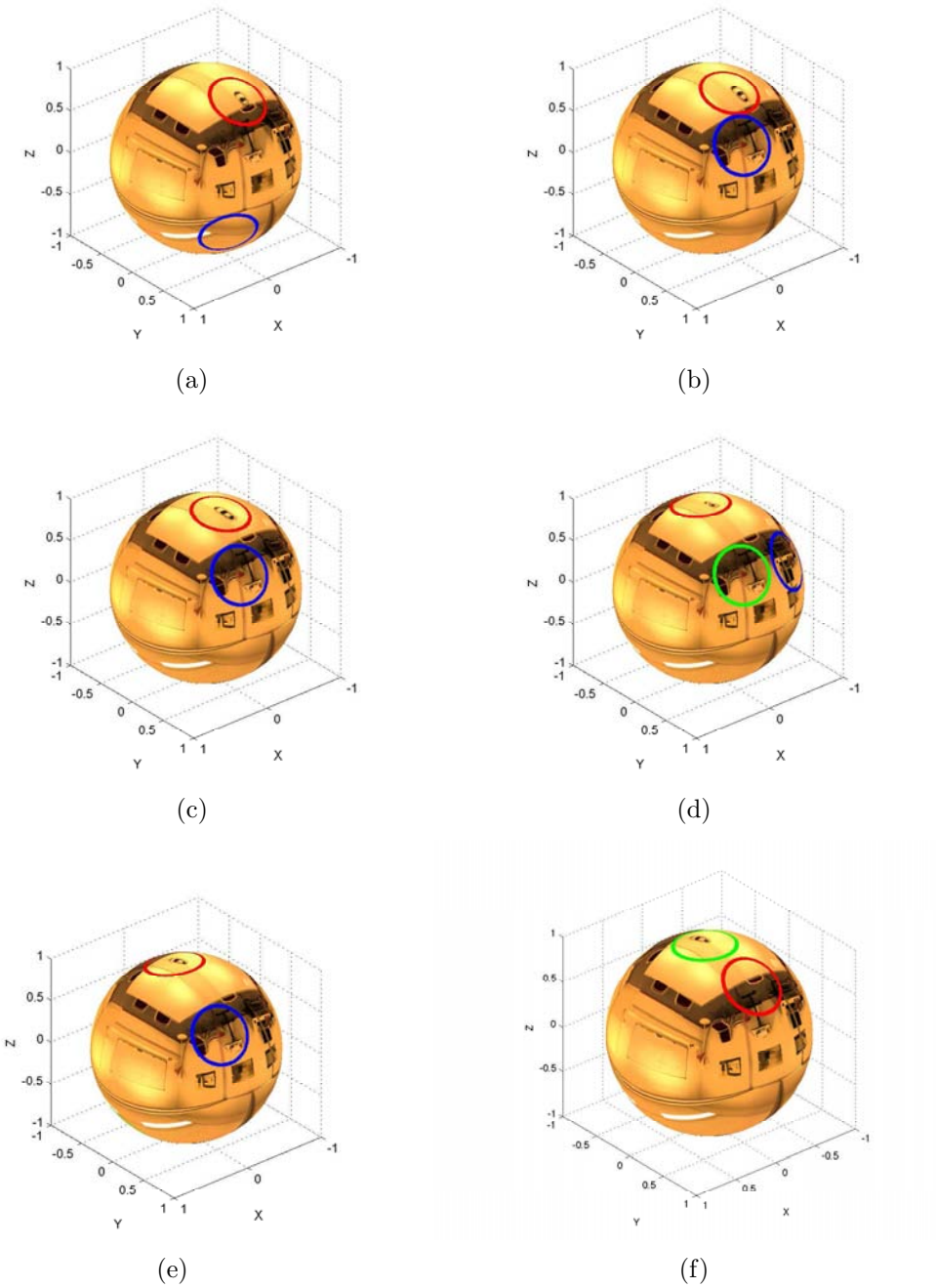
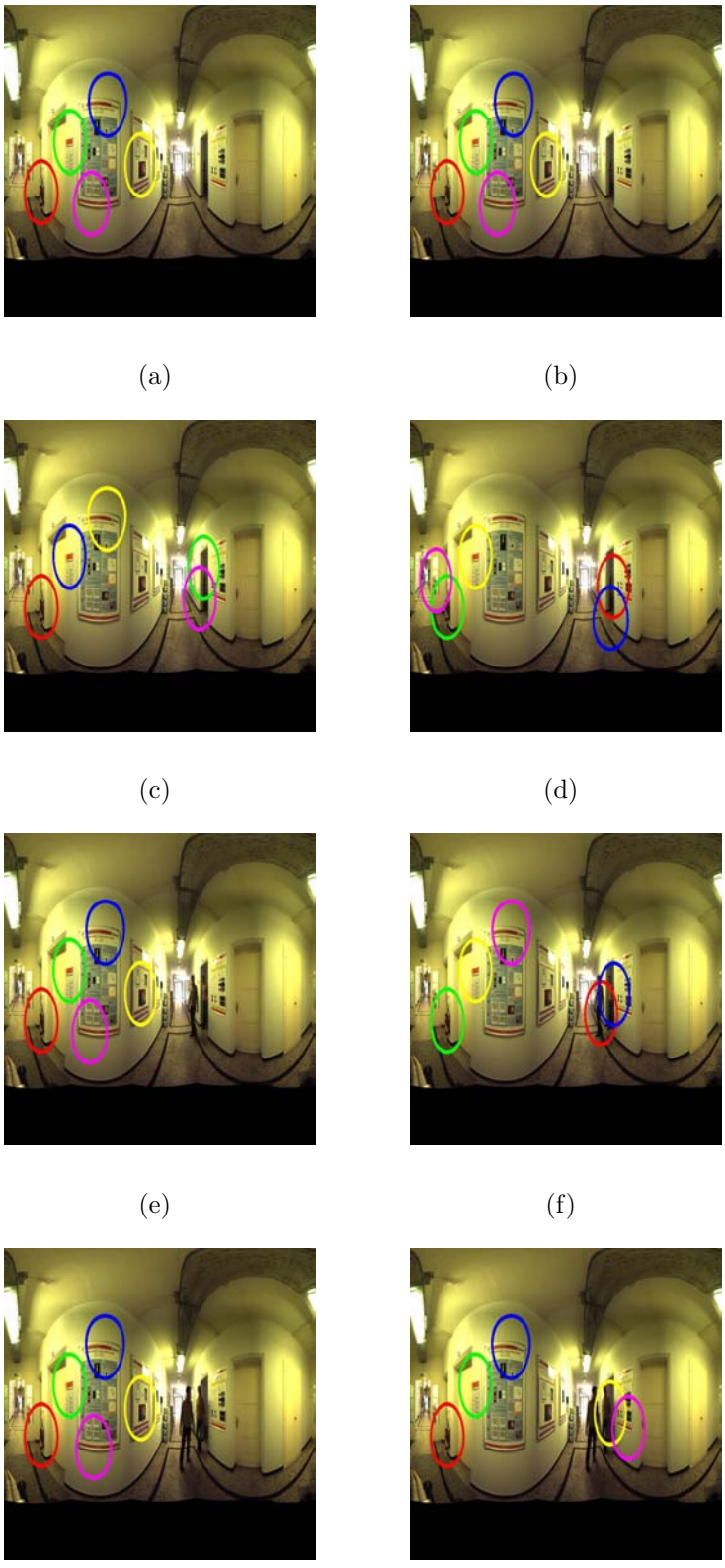


Figure 5. Spots of attention in some of the frames in the spherical image sequence: (a) frame 6; (b) frame 8 ; (c) frame 10 ; (d) frame 12; (e) frame 14; (f) frame 16. The most salient object in this image sequence is given in red. The motion is dominant in determining the dynamic spherical saliency.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 6. Spots of attention in still-background omnidirectional video. For computing the spherical motion conspicuity, only the magnitude (on the left) and only the phase (on the right) is considered.

from the color opponency even a motion is available in scene. Throughout the sequence, it is the red object that is constantly the most salient location. When a motion appears Figure 6(c), it determines the second spot in the corresponding frame but immediately afterwards none of the two motions determine any saliency (Figure 6(e,g)). On another hand, when the spherical motion contrast in phase is considered, the most salient object is the walking person Figure 6(d) and it remains as such in the following frames where there is a motion Figure 6(f,h). Even-though the rank of the attention spots changes in (h), the motion remains dominant. In conclusion, what concerns still background spherical image sequences, the motion phase highly influences the final saliency on the sphere.

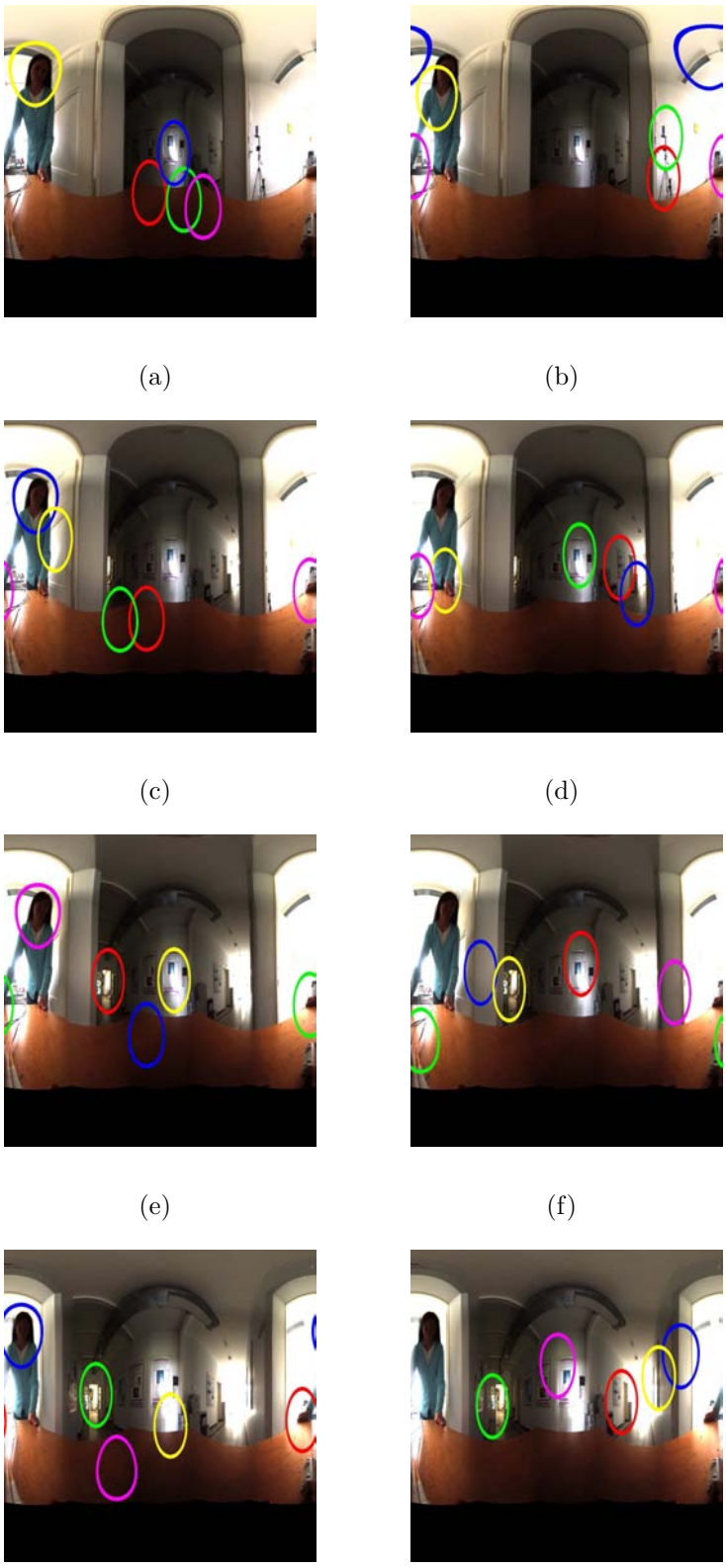
The second experiment is carried out on moving background omnidirectional video. The sequence is obtained while the sensor is placed on a moving table. In the same time, there is a walking person who moves in parallel to the table. The same reasoning for computing the spherical motion conspicuity is considered, i.e. the motion magnitude and phase are examined separately. Again, the unwrapped spherical version is shown on Figure 7 with the first five spots of attention. In this figure on the left, the magnitude is considered while on the right - only the phase is used for computing the spherical motion cue.

It is interesting to note that in this case, when the motion contrast in magnitude is considered, the face of the person is always detected as salient even with a different rank (Figure 7(a,c,e,g)) while this is not the case when only the spherical motion phase is used. On the other hand, the majority of the spots are in the background when the contrast is in the phase (Figure7(b,d,f,h)).

5.2. Spherical versus Euclidean Dynamic Visual Attention

In this section we perform a simple experiment using the "car" omnidirectional sequence of spherical images. In figure 8 are shown the spots of attention as detected by the spherical VA algorithm and present them in their unwrapped version, where $\varphi \in [0, 2\pi)$ is at the horizontal axis and $\theta \in [0, \pi]$ is at the vertical axis. Let us remind that the beginning and the end of the vertical axis correspond to the North and the South pole of the sphere, respectively. Frames 6, 8,10,12,14 are shown with the corresponding first three spots of attention detected by applying the VA in spherical coordinates. The spots look distorted in this unwrapped version while they represent a disk. For instance, the red spot tends to a line while in spherical coordinates it is a disk approaching the sphere's pole. We can see, that as expected, the car is detected as the most salient object (in red) along the sequence while moving even when it is passing through the north pole of the sphere (Figure 8(c),(d),(e)). After the car stops, the most salient spot is the red chair (Figure 8 (f)). In fact, when there are no moving objects in the scene, the static spherical saliency map is dominant in the context of computing the final spherical saliency map.

Let us now, consider the unwrapped spherical image as if it were an Euclidean one and apply on it the Euclidean dynamic visual attention model. The Euclidean motion cue is obtained using both the magnitude and phase of the motion vector. The motion is detected using block-matchng algorithm as well. The obtained results are depicted in Figure 9. In fact, the car is completely distorted while passing through the north pole. Here, we refer to a distortion, as any deformation in the scene resulting from unwrapping of the spherical image. Such distortion would persist even in the case of



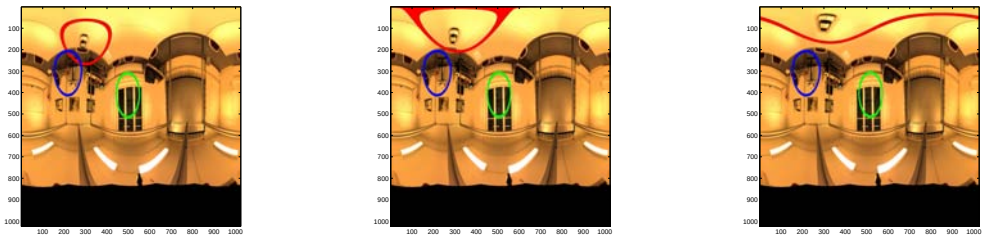
(a) (b)

(c) (d)

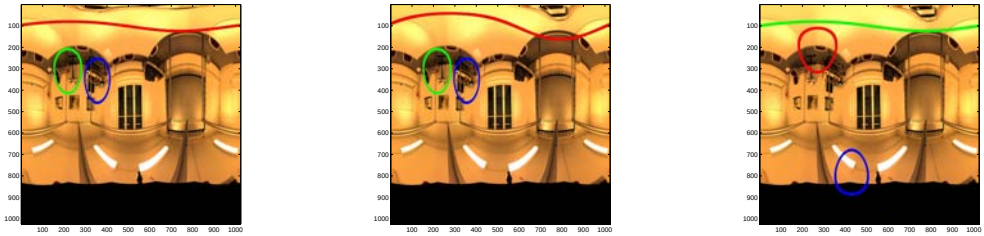
(e) (f)

(g) (h)

Figure 7. Spots of attention in moving background omnidirectional video. Left: motion magnitude. Right: spherical motion phase.

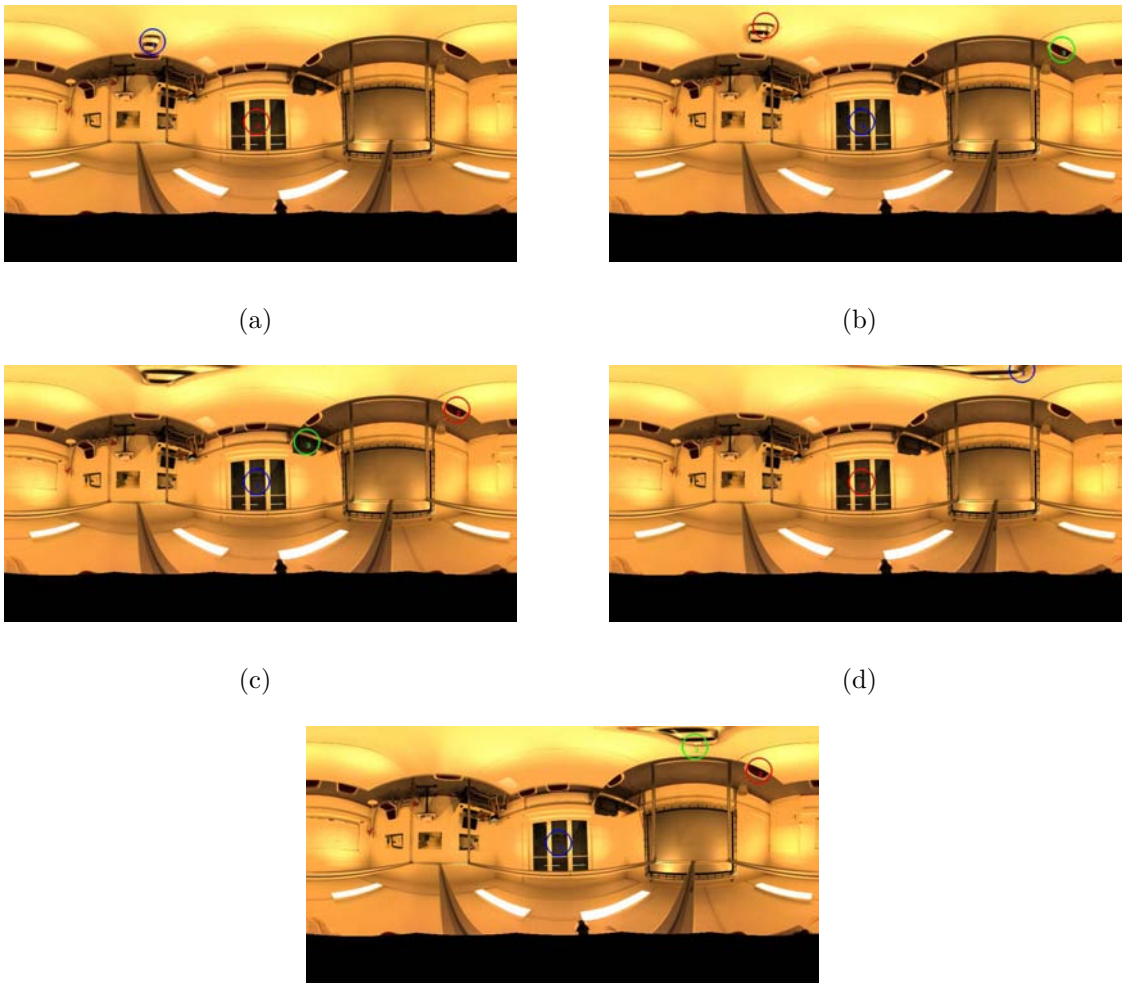


(a) (b) (c)



(d) (e) (f)

Figure 8. Euclidean vs Spherical dynamic VA: the spots of attention are detected by the VA on the sphere and the first three detected objects are shown. While moving even through the north pole of the sphere, the car is the most salient object in this sequence. In the last frame (f), the most salient object is a red chair since the car is not moving anymore



(a) (b)

(c) (d)

(e)

Figure 9. Euclidean vs Spherical dynamic VA: the Euclidean VA algorithm is applied on the unwrapped omnidirectional sequence. The first three most salient locations are shown and ranked. Even the car is the only moving object in the scene it is detected as the most salient location only in frame 6 (a) and partly in frame 12 (d).

mapping any omnidirectional image to a cylinder, i.e. creating a panoramic version of the omnidirectional image. Even-though, the car is the only moving object in the scene it is detected as the most salient location only in frame 6 (a) and partly in frame 12 (d). While it is completely distorted on the north pole, it is not perceived as a unique object from Euclidean point of view and this is the main reason it is not detected as salient object at all.

In conclusion, through this experiment we have demonstrated that the dynamic spherical VA algorithm performs better in omnidirectional images compared to the standard (Euclidean) VA algorithm.

CONCLUSIONS

In this paper we have proposed an algorithm for computing the spherical motion conspicuity map. It is based on extracting the motion vector between two consecutive spherical images where block matching in spherical coordinates is performed. Then, the dynamic spherical saliency map has been computed using the motion conspicuity map. The dynamic saliency map on the sphere is obtained by fusing together the static and motion spherical saliency maps. Furthermore, this algorithm has been applied on a real sequence of spherical images and the spots of attention based on the final spherical saliency map have been detected. The spherical motion magnitude and phase have been examined separately while using spherical video with still and moving background. Finally, a comparison of the dynamic spherical spots detection versus the Euclidean spots detection has been done and thus the advantages of the dynamic VA on the sphere against the Euclidean dynamic VA have been confirmed. Furthermore, the proposed algorithm is universal in omnidirectional image sense because it can be applied on different kind of such images once they have been mapped on the sphere.

REFERENCES

1. Watanabe et al. Attention-Regulated Activity in Human Primary Visual Cortex, J. Neurophysiology,vol.79, pp. 2218-2221, 1998.
2. Koch, Ch. and Ullman, S., Shifts in Selective Visual Attention: Towards the underlying Neural Circuitry, Human Neurobiology, vol.4, pp. 219-227, 1985.
3. L. Itti, Quantifying the contribution of low-level saliency to human eye movements in dynamic scenes, Visual Cognition, vol.12, pp. 1093-1123, 2005.
4. A. Bur, P. Wurtz, R.M. Muri, and H. Hügli, Motion integration in visual attention models for predicting simple dynamic scenes, Human Vision and Electronic Imaging XII, vol. 6492:49219, 2007.
5. L Demanet and P. Vandergheynst, Gabor Wavelets on the Sphere, Proc. SPIE Wavelets X conf.,San Diego, USA, 2003
6. O. Le Meur, and P. Le Callet, and D. Barba, and D. Thoreau, Predicting Visual Fixations on video based on low-level visual features, Vision research, vol. 47, pp. 2483-2498, 2006
7. J.K. Tsotsos, and Y.J. Liu, and J.C. Martiniez-Trujillo, and M. Pomplun, and E. Simine, and K. H. Zhou, Attending to visual motion, CVIA, vol.100(1-2), pp.3-40, 2005

8. C. Geyer and K. Daniilidis, Catadioptric projective geometry, IJCV, vol.45(3), pp. 223-243, 2001.
9. LADYBUG, http://www.ptgrey.com/products/spherical.asp.
10. I. Bogdanova and A. Bur and H. Hügli, Visual Attention on the Sphere, IEEE Transactions on Image Processing, vol.17(11), pp. 2000-2014,2008.
11. I. Tosic, and I. Bogdanova, and P. Frossard, and P. Vandergheynst, Multiresolution Motion Estimation for Omnidirectional Images, Proc. EUSIPCO, 2005.