Towards Space-Time Semantics In Two Frames

Karla Brkić *[1], Axel Pinz[2], Zoran Kalafatić[1], and Siniša Šegvić[1]

[1] University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia
[2] Graz University of Technology, Austria

Abstract. We present a novel, low-level scheme to analyze spatial and temporal change within a local support region. Assuming available region correspondences between two adjacent frames, we divide each region into a regular grid of patches. Depending on the change of an image function inside the patch over time, each patch is assigned weights for the following four labels: "C" for a constant patch, "O" when new information originates from outside the support region, "I" for "inner" changes, and "N" for information from neighboring patches. Our method goes beyond optical flow, as it provides an additional semantic level of understanding the changes in space-time. We demonstrate how our novel "COIN" scheme can be used to categorize local space-time events in image pairs, including locally planar support regions, 3D discontinuities, and virtual vs. real crossings of 3D structures.

1 Introduction

In this paper we focus on discovering scene structure at a very low level (e.g. the level of an interest point or a small image patch), assuming that a temporal sequence of at least two frames is available. Changes in low-level appearance over the course of two frames are encoded in a semantically meaningful descriptor which assigns weights to the following hypotheses: constant, influenced by neighbor, inner change, outer change. Using the descriptor, we are able to reason about coplanarity, discontinuity and general stability of appearance of the region of interest.

Analysis of low-level dynamic structure in video is related to several research strands. One is spatiotemporal texture recognition. There, short video clips are usually represented by generative statistical models [1]. A recent approach [2] attempts to capture the spectrum of dominant image motions which arise for instance when looking at the snow falling in the wind. Each video clip is represented by a 28 bin histogram containing evidence for a small predefined set of image motions, such as e.g. "upward motion of two pixels per frame".

Another related research strand includes the analysis of occluding boundaries and video segmentation [3]. Many of the existing approaches are based on locating contrasting optical flow (e.g. [4, 5]). A promising approach which avoids the

* This research has been supported by the University of Zagreb Development Fund and Research Centre for Advanced Cooperative Systems (EU FP7 #285939).

dependence on optical flow has been presented [6]. There, the occluding boundaries are detected as regions with high curvature of spatio-temporal contrast in the frequency domain, much along the lines of the known Harris corner detector.

Finally, our work is related to sparse spatio-temporal features [7–12]. An early treatise of this concept [7], proposes an interest point detector suitable for classifying actions such as eye opening or knee bending. A more advanced implementation of that concept has been proposed in [13], as a generalization of the known HOG detector to the 3D case. A recent approach [14] has demonstrated that sparse spatio-temporal features can be used to detect occlusion boundaries as well.

In this work, we analyze changes in local appearance by dividing the region of interest into a grid of patches and assigning weights to individual hypotheses based on patch histograms in two consecutive frames. We avoid building on top of optical flow, as dense optical flow can not be accurately computed in many cases of practical importance. Optical flow approaches typically optimize some kind of correspondence criterion: if the objective function is multimodal (such as at an aliased texture) it is easy to get stuck at a wrong local maximum. In contrast to the work of Derpanis and Wildes, [6, 2] our approach supports the notion of emergent events of the low level structure in video, and can naturally represent occurrences of uncovered texture behind the occlusion boundaries.

2 The COIN descriptor

The COIN descriptor is a semantic descriptor of temporal change in local appearance of an image region. The observed region is divided into a regular grid of $m \times n$ patches. The semantic description is obtained by studying patch appearance in two consecutive frames at times t_1 and $t_2 > t_1$.

First, we model patch appearance by histograms of a particular image function. Then, the histogram of each patch in t_2 is compared to the histograms of the patch itself and the 4-neighborhood in t_1. Based on the comparison and the position of the patch itself within the grid, we assign weights to one of the following four hypotheses: C - the patch remained constant in time, O - there is new information in the patch which originated from outside the grid, I - the patch changed "by itself", i.e without outer influence, N - the information in the patch originated from one of its neighbors from the previous frame.

COIN can be calculated for various spatial resolutions in three different scenarios: i) using a fixed grid over entire frames, we want to find semantically interesting structures in a scene; ii) on the level of an interest point, we wish to describe an interest point and answer questions such as is it stable, does it depict a depth discontinuity or a planar surface; iii) on the level of an object of interest, we search for spatio-temporal structure within a bounding box. While case i) is "camera-centered", cases ii) and iii) require frame-to-frame correspondences, established by interest point or object tracking. In this paper we restrict the experimental analysis of the descriptor properties to the level of an interest

point, but the descriptor is well-suited to be applied to scenarios i) and iii) as well.

2.1 Calculating patch histograms

Without loss of generality, let us assume that we have two bounding boxes around an image region of interest, one taken at time t_1 and another at time $t_2 > t_1$. We denote their widths and heights as (w_1, h_1) and (w_2, h_2), respectively. Each bounding box is divided into a regular grid of $m \times n$ patches. The size of an individual patch within the first boundig box is $(w_1/m, h_1/n)$, while the size of an individual patch within the second bounding box is $(w_2/m, h_2/n)$. We denote the patches within the grid at time t_i as $p_1^{t_i}, p_2^{t_i}, \ldots, p_{m\times n}^{t_i}$. For an illustration, see Fig. 1, depicting COIN calculation on a series of synthetic images. The images, shown in the first row of the figure, consist of a regular grid of diversely colored rectangles. In dividing the images into a regular grid of 5×5 patches, each patch will fall perfectly into one colored rectangle, and will contain a single dominant hue plus the black rectangle border.

For each patch $p_i^{t_1}$ we calculate a patch histogram $H_i^{t_1}$ for a previously chosen image function (hue, value, saturation, or gradient). The same is done for patches $p_j^{t_2}$. This step results in two collections of a total of $m \times n$ histograms (which we call grids of histograms), one obtained in time t_1 and another in time t_2. Each histogram represents a $(w/m) \times (h/n)$-sized subregion of the image ROI. We denote the histogram of i-th patch in time t_j as $H_i^{t_j}$.

After calculating image function histograms, they are normalized so that $\sum b_k = 1$, $b_k \in H_i^{t_j}$, i.e. the values of all histogram bins sum to 1.

2.2 Assigning C, O, I, N hypotheses

Having calculated an image function histogram of a patch in time t_2, we want to determine what happened to the patch, given its appearance in t_1. To do so, we use three operations: histogram intersection, $H_i \cap H_j$; histogram subtraction, $H_i \backslash H_j$; and histogram mass computation, $\mu(H_i)$. The result of histogram intersection is a new histogram in which the value of each bin is the minimum value from the two intersected histograms. Subtraction is achieved simply by subtracting bin values, and mass computation by summing them.

We adhere to the normalized histograms paradigm adopted so far, and aim at assigning each patch $p_k^{t_2}$ a four-dimensional, COIN function (C, O, I, N), which is normalized, so that $C+O+I+N = 1$. The idea is to first calculate the change in the appearance of the patch histogram which cannot be attributed neither to its appearance in the previous frame nor to one of its neighbors. This change is then labeled as inner or outer, and the remaining mass is distributed to constant and neighbor hypotheses. We start by finding the patch $p_l^{t_1}$ among the patches in 4-neighborhood of $p_k^{t_2}$ for which $\mu(H_k^{t_2} \cap H_l^{t_1})$ is maximal. In other words, for the patch in the current frame, we find the most similar 4-neighbor in the previous frame based on the mass of intersection of their histograms. Note that here we

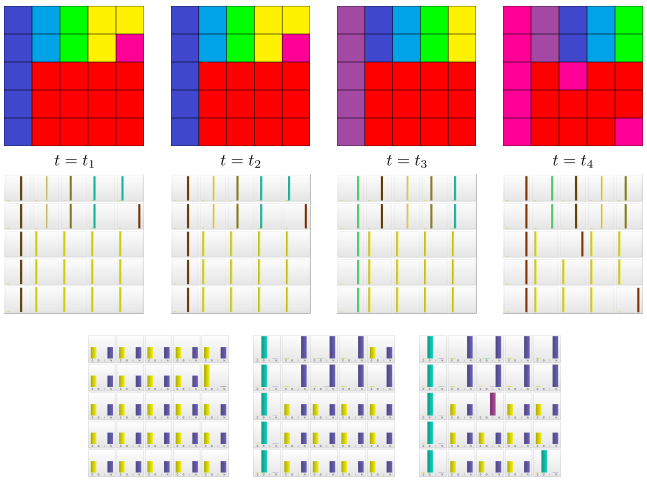

Fig. 1: Our running example explaining all the steps of the COIN calculation process. The descriptor components for each image patch are displayed in different colors and in order: C, O, I, N. C occurs both in transition from frame 1 to frame 2 (they are equivalent), as well as in the red area of the grid in the first three frames. O occurs in the left column of frames 3 and 4. In frame 4, two red patches change to magenta. Magenta was not seen within the grid before, therefore we assign O to the bottom right patch and I to the center patch.

observe only the 4-neighborhood, assuming that we have sufficient spatial and temporal resolution to capture interesting events under that constraint. Next, by intersecting the current and the previous patch histogram, we get what these two histograms have in common. If we then subtract that from the current patch histogram,

$$\Gamma = H_k^{t_2} \backslash (H_k^{t_1} \cap H_k^{t_2}), \tag{1}$$

what remains is the histogram of change in appearance of the patch. This change either arrived from a neighbor, or the patch was changed by inner/outer influences. We can now compute the joint weight of inner and outer hypotheses,

$$\omega_{IO} = \Gamma \backslash (\Gamma \cap H_l^{t_1}), \tag{2}$$

following the same logic as in the computation of Γ.

The value $\mu(\omega_{IO})$ is the mass of the total change in appearance of the histogram which cannot be attributed to a neighbor. There are two cases that can happen: (i) there is information which was not seen before, and it originated within the patch itself, or (ii) there is change which originated from outside the bounding box. To distinguish between these cases, we use a prior: a $m \times n$ matrix of values from 0 to 1, where the value α_k at the position of the studied patch k denotes the probability of the inner hypothesis. We usually choose values of α_k to reflect that outer change is more likely to occur closer to the border of

the bounding box than inner change. We might, for instance, consider using a Gaussian kernel or a similar matrix. The values of I and O are then:

$$I = \alpha_k \omega_{IO}, \quad O = (1 - \alpha_k)\omega_{IO} \tag{3}$$

As ω_{IO} is always smaller than 1, we can find the rest of the mass which is to be assigned to constant and neighbor hypotheses as:

$$\omega_{CN} = 1 - \omega_{IO} \tag{4}$$

We denote the intersection mass between the histogram of the current patch and the histogram of the previous patch as

$$\omega_C = \mu(H_k^{t_2} \cap H_k^{t_1}), \tag{5}$$

and the intersection mass between the histogram of the current patch and the histogram of the most similar neighbor as

$$\omega_N = \mu(H_k^{t_2} \cap H_l^{t_1}) \tag{6}$$

Both ω_C and ω_N are between 0 and 1. We would like to proportionally distribute the remaining mass ω_{CN} to ω_C and ω_N to obtain the weights of C and N hypotheses. We do so in the following manner:

$$C = \frac{\omega_C}{\omega_C + \omega_N}\omega_{CN}, \text{ and } N = \frac{\omega_N}{\omega_C + \omega_N}\omega_{CN} \tag{7}$$

Following the computation outlined above, we obtain the values C, O, I, N which form the COIN function of the patch $p_k^{t_2}$. By concatenating values of COIN functions for each patch in the grid, we obtain the COIN descriptor. As an example, consider the first two frames in Fig. 1. The two frames are exactly the same. Calculated COIN descriptors between frames are shown in the third row of the figure. They are represented as $m \times n$ 4-bin histograms, where each histogram represents the weights C, O, I, and N. Due to the nature of COIN computation, all weights sum to 1. As the first two frames are equal, we might expect $C = 1$ for all patches of the COIN descriptor, but this is the case just for histogram 10. Notice that every patch except patch 10 has a twin neighbor, i.e a neighbor which looks exactly like it. Therefore, it might be the case that the patch and the neighbor have switched places, even though apparently the image is unchanged. Thus, all patches but patch 10 have $C = N = 0.5$. Patch 10 has no twin neighbors, resulting in $C = 1$. In the two top rows for transitions from frames 2 to 3 and 3 to 4 we see $N = 1$. This is due to colored patches moving to the right. Finally, we see inner and outer change in the second and the third COIN descriptor. In the second COIN, we see that patches near the left edge bring in new information. In the third COIN, we additionally see $I = 1$ in the center patch, outer change in the bottom right corner. Here, we used a simple prior of $\alpha_k = 0$ for all the patches at the border of the bounding box, $\alpha_k = 1$ otherwise. Of course, in real scenarios inner change could also happen at the border of the bounding box (a person suddenly appearing in a window, or a semaphore changing its signal), but this should then be covered by an adequate selection of the prior.

Fig. 2: An 8×8 grid superimposed on frames 3 and 4 and the calculated COIN descriptor.

3 Experiments

In our experiments, we highlight and analyze various properties of our novel COIN descriptor. We start (3.1) with some more analysis on the synthetic images used in section 2 regarding parameter settings (i.e. grid size). Next (3.2), we present results on real image sequences taken in our lab. In these experiments we concentrate on space-time semantics that can be deduced from COINs. To highlight these semantic aspects of COINs, we have chosen a colorful world that can be well represented by histograms of hue. Finally (3.3), we show results in discriminating different types of low-level structure on a sequence from the DTU Robot Data Set.

3.1 Further experiments on the synthetic image sequence

In section 2 we used a simple synthetic sequence as our running example. The sequence consists of colored rectangles, and in our previous discussion we used a grid of 5×5 patches, which aligned perfectly with the grid present in the image. But what if we were to use a finer grid, where patches would contain multiple colors? Would the underlying behavior still be visible? Fig. 2 illustrates a grid of 8×8 patches superimposed on synthetic images in frames 3 and 4, and the COIN descriptor calculated on that grid. All other parameters, including the inner prior, were the same as in the previous example. What is apparent from this simple experiment is that the COIN descriptor generates a semantically correct description of what happens in time even if we use unfavorable grid sizes. In the first column of the 8×8 grid we clearly see outer change, the same as in the first column of the 5×5 grid (see Fig.1). We can also notice that both the inner and the outer change of red to magenta are still reflected within COIN histograms, although now distributed through four cells in the grid instead of one. This is because magenta rectangles now span through four image patches, insted of corresponding perfectly to a single patch. It is interesting to notice the first two rows within the fifth column of the histogram grid. There, we would expect to see maximum weight assigned to the neighbor hypothesis, but it is instead assigned to the inner and outer hypotheses. If we have a look at the fifth patch in the first row in frame 1, it is mostly light-blue, with a little bit of green. Its first neighbors are either light blue or green, the nearest dark blue neighbor

is 2 places in the grid away. In frame 2, the patch receives a lot of dark blue. As we analyze only the 4-neighborhood of the patch, the neighbor hypothesis gets weighted down and the outer hypothesis dominates, which is a reasonable behavior. If undesirable, it could be tuned out by modifying the parameter α_k.

3.2 The color world experiments

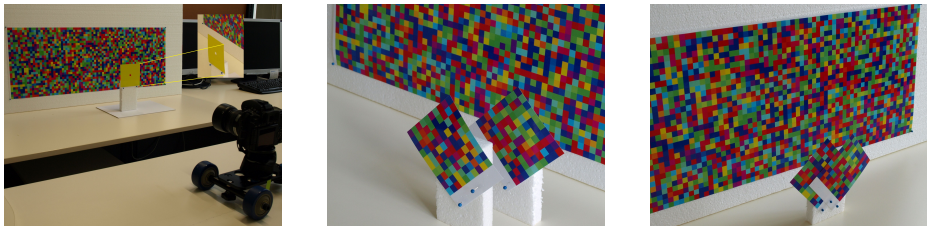

Fig. 3: Our experimental setup. (left): Camera taking a sequence. (middle and right): Our experimental setup to capture the "real" and "virtual" crossings shown in Fig. 4. The camera is translated parallel to the background target at a distance of 1m, using a focal length of 100 mm and an aperture setting of 22. All targets show random hue patches of maximum saturation and value, foreground targets are 10×10 cm. For the "virtual" crossing (Fig. 4 top), the two targets are placed at 22 cm, and 40 cm in front of the background, while the"real" crossing (Fig. 4 bottom) is placed 15 cm in front of the background.

Verifying that the COIN concept works on simple synthetic images, we moved on to testing it on real images. For that purpose, we created the experimental setup depicted in Fig. 3. The background is an 80×30 cm poster comprised of 1×1 cm squares. While all squares have maximum saturation and value in HSV space, the hue of each square is random. In the foreground, we have placed several arrangements of 10×10 cm targets, including homogeneous as well as random colors, holes in a target, and real as well as "virtual" crossings. In all cases, the camera has been translated from right to left, frontoparallel to the background, at a distance of approx. 1 m. In these experiments, the observed behavior of COIN descriptors was very similar to its behavior on synthetic data. Due to space limitations, here we will briefly present just a descriptor of a real crossing and a descriptor of a virtual crossing and discuss how the two might be distinguished.

The top row of Fig. 4 depicts two frames of two textured targets overlapping, and the corresponding COIN descriptor calculated at the point where the targets cross. The overlap is virtual, meaning that in 3D the targets are apart. In contrast, the bottom row of Fig. 4 depicts a case where the targets are touching. Notice that in the case of the virtual crossing the front target is actually occluding the back target as time advances. This means that we would see a lot more

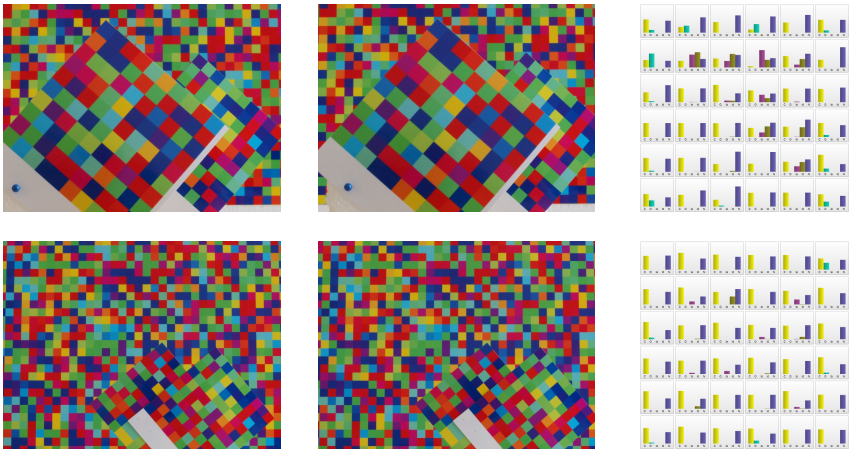

Fig. 4: Two targets which do not overlap in 3D ("virtual crossing", top) and two targets which do overlap in 3D ("real crossing", bottom), along with the corresponding COIN descriptors built around the crossing. The COINs were built over a 6×6 grid, using 10-bin hue histograms.

inner change if we *reversed* the sequence. To illustrate, we've included an additional column in our representation of COIN descriptors, which we named I_d, for "inner disappearing". I_d is equal to I of the COIN of the reversed sequence. Our regular I was renamed to I_a, for "inner appearing". In the case of a real crossing, we see dominating constant/neighbor hypotheses at the edge where the targets cross, while in the case of a virtual crossing, we notice the appearance of inner/outer change, which is, as expected, especially prominent with I_d. Based on this finding, we propose a simple measure of the level of spatio-temporal discontinuity:

$$\Sigma_I^{t_2} = \sum \frac{I_a + I_d}{2} \tag{8}$$

where I_a and I_d are calculated for each patch $p_k^{t_2}$ within the grid at time t_2.

3.3 Experiments on the DTU Robot Data Set

Having found that the COIN descriptor performs well on artificial scenes, we tested whether it can discriminate between 3D discontinuities, coplanarities and virtual crossings. To the best of our knowledge, there are not any publicly available datasets with suitably or similarly labeled video data. However, the recent DTU Robot Data Set by Aanaes et al. [15] consists of 60 scenes and scene surface data obtained from a structured light scan, so it has the potential for automated groundtruth acquisition.

In the experiment reported here, we used the first 49 frames of scene number 14 from the DTU Robot Data Set. We manually labeled ten interesting structures

Fig. 5: Points selected for manual experimental validation of COIN from the DTU Robot Data Set (#014, frame number 24).

Table 1: Experimental results on classifying point types on the DTU Robot Data Set sequence #14. Point IDs are the same as in Fig. 5

	small discontinuity			large discontinuity			virtual crossing		coplanarity	
ID	3	4	7	1	8	9	6	5	2	10
$\overline{\Sigma_I}$	0.20	0.19	0.23	0.62	0.63	0.52	0.71	0.39	0.00	0.04

in the scene throughout all 49 frames: three small depth discontinuities, three large depth discontinuities, two coplanarities and two virtual crossings, as shown in Fig. 5. Then, COIN descriptors were built using an offset of 5, i.e. between frames 1 and 6, 6 and 11 etc. We used a grid of 5×5 patches and histograms of hue with 10 bins. As a measure of the level of discontinuity within a grid, we used Σ_I (Eq. 8). Table 1 shows the average value of this measure for four different categories of points. We can notice that the smaller and and the larger depth discontinuities as well as coplanarities seem to be well separated by the value of this measure. One virtual crossing has a very large value of $\overline{\Sigma_I}$, while the other has a value in between the values for small and large discontinuities. This is correct, because a virtual crossing is by definition a depth discontinuity. It remains to be seen on a larger dataset whether we will be able to distinguish virtual crossings from other discontinuities using Σ_I only, or whether we will need other more complex measures which would, for instance, take into account the position of the inner change within the grid.

4 Conclusion and outlook

We have presented a quite novel scheme to describe and detect semantics of local frame-to-frame appearance change. Aiming to extend the success of histogram / appearance based methods in 2D to space-time, we devised a solid mathematical framework for reliable computation of weights assigned to C, O, I, N hypotheses. The applicability of the descriptor was confirmed by our experiments, which have demonstrated how to use COIN for reliable labeling of coplanarity vs. discontinuity and to distinguish between real and virtual crossings.

We plan to investigate the properties of COIN on a larger amount of data, which will be obtained either by automated groundtruth acquisition or manual labeling. We are also interested in using COIN to distinguish other kinds of local structures, e.g. convexities or concavities. Using COINs inside bounding boxes of more complex image events, e.g. articulated object motion, should provide excellent means to classify complex motion patterns. Furthermore, aggregating COINs to sequences of COINs over video might provide a higher semantic level.

References

1. Doretto, G., Chiuso, A., Wu, Y., Soatto, S.: Dynamic textures. Int. J of Comp. Vis. **51** (2003) 91–109
2. Derpanis, K., Wildes, R.: Spacetime texture representation and recognition based on a spatiotemporal orientation analysis. IEEE TPAMI (2012)
3. Doulamis, A., Doulamis, N., Ntalianis, K., Kollias, S.: An efficient fully unsupervised video object segmentation scheme using an adaptive neural-network classifier architecture. Trans. Neur. Netw. **14** (2003) 616–630
4. Ogale, A.S., Fermüller, C., Aloimonos, Y.: Motion segmentation using occlusions. IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005) 988–992
5. Stein, A.N., Hebert, M.: Local detection of occlusion boundaries in video. Image Vision Comput. **27** (2009) 514–522
6. Derpanis, K., Wildes, R.: Detecting spatiotemporal structure boundaries: Beyond motion discontinuities. Proc. ACCV (2009) 301–312
7. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: VS-PETS. (2005)
8. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: Proc. ICCV. Volume 1. (2005) 166 – 173
9. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proc. ICCV. (2003)
10. Laptev, I.: On space-time interest points. Int. J. of Comp. Vis. **64** (2005) 107–123
11. Le, Q.V., Zou, W.Y., Yeung, S.Y., Ng, A.Y.: Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In: Proc. CVPR. (2011) 3361–3368
12. Wang, H., Ullah, M.M., Kläser, A., Laptev, I., Schmid, C.: Evaluation of local spatio-temporal features for action recognition. In: Proc. BMVC. (2009) 127
13. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: Proc. BMVC. (2008) 995–1004
14. Brkić, K., Pinz, A., Šegvić, S., Kalafatić, Z.: Histogram-based description of local space-time appearance. In: LNCS 6688. (2011) 206–217
15. Aanæs, H., Dahl, A., Steenstrup Pedersen, K.: Interesting interest points. Int. J. of Comp. Vis. **97** (2012) 18–35