Neural Processing Letters (2005) 22:125–147
DOI 10.1007/s11063-005-2158-1

Motion-Driven Segmentation by Competitive Neural Processing

SONIA MOTA*, EDUARDO ROS, JAVIER DÍAZ, EVA M. ORTIGOSA and ALBERTO PRIETO
Departamento de Arquitectura y Tecnología de Computadores, E.T.S.I. Informática, Universidad de Granada, Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain. e-mails:{smota, eros, jdiaz, emartinez, aprieto}@atc.ugr.es

Abstract. Bio-inspired energy models compute motion along the lines suggested by the neurophysiological studies of V1 and MT areas in both monkeys and humans: neural populations extract the structure of motion from local competition among MT-like cells. We describe here a neural structure that works as a dynamic filter above this MT layer for image segmentation and takes advantage of neural population coding in the cortical processing areas. We apply the model to the real-life case of an automatic watch-out system for car-overtaking situations seen from the rear-view mirror. The ego-motion of the host car induces a global motion pattern whereas an overtaking vehicle produces a pattern that contrasts highly with this global ego-motion field. We describe how a simple, competitive, neural processing scheme can take full advantage of this motion structure for segmenting overtaking-cars.

Key words. low-level vision, bio-inspired system, neuronal motion detection, overtaking-car segmentation

1. Introduction

Motion processing is an important function for the survival of most living beings and so their visual systems have specific areas dedicated to this task alone [1]. Primary visual areas are modelled using space–time receptive filters [2–4] to compute motion as suggested by neuro-physiological data [5].

Simoncelli and Heeger (S&H) modelled how the cortical areas (V1 and MT cells) can extract the structure of motion through local competitive neural computation [4, 6]; their results were in accordance with neuro-physiological data. The output layer produces neural velocity population coding, which is inefficient compared with more mathematically based algorithms, but represents an advantage if the post-processing is done through neural computation, as we describe in this paper. MT cells are highly sensitive to a very specific direction and speed of movement. This characteristic depends on the cortical layers being highly interconnected. Hence MT activity represents smooth and homogeneous motion patterns.

*Corresponding author.

We propose a post-processing structure that takes advantage of these properties. Motion estimation based on local operators is normally very noisy and requires further post-processing before any segmentation can be addressed. We describe here how a simple connectivity pattern facilitates the neural computation of noisy motion information. This connection pattern makes individual cells behave as dynamic filters that are sensitive to more reliable movement features than simple space–time correlations. This post-processing layer is composed of cells that collect the output activity from MT cells sensitive to similar motion primitives. We also describe how this can enhance the capability of segmenting rigid-body motion by connecting MT cells of local neighbourhoods throughout the visual field.

Motion-driven scene segmentation using neural networks has been addressed by several authors [7–9]. These approaches use motion estimation maps (optic flow) as inputs and apply neural networks to classify motion patterns. In this case neural networks are applied to achieve efficient classification of noisy patterns. The learning and generalization capabilities of neural networks make them particularly interesting for this kind of application.

There are also motion extraction schemes based on local neural processing [10–12]. In this case no learning is applied; the key element that motivates these approaches is that they are based on local neural processing, which facilitates their hardware implementation for real-time processing and smart-sensor development [10,13–16].

From a completely different perspective, Simoncelli proposed a biologically inspired motion-processing scheme that tries to explain how motion is processed by local elements in different visual areas in the brain [4,6,17]. Although the initial motivation of Simoncelli's approach was to study and simulate how motion information is processed in cortical areas, this motion estimation scheme achieves a similar performance rate to other classical approaches [18–20]. Nevertheless, Simoncelli's scheme is not often used because of its computational complexity on standard computing platforms (conventional PCs). As it is based on local processing and population coding, however, its physical implementation on specific hardware may be interesting if its inherent massive parallelism can be exploited. Although the hardware implementation of this approach is not described in this paper its feasibility has been evaluated (This is commented on briefly in Section 4.). Following Simoncelli's ideas on how motion is processed by the brain, we focus on one step beyond the motion-extraction stage. We propose a neural structure that takes full advantage of the inherent characteristics of Simoncelli's approach: local neural processing and population coding. The proposed neural processing scheme is based on competitive computation among a layer of neurons that collects the motion information. This post-processing stage efficiently cleans up spurious noise patterns, and filters only coherent spatio-temporal patterns. These patterns emerge naturally from the population coding generated by the Simoncelli approach. We describe here how simple on-centre-off-surround connectivity patterns can form a simple architectural primitive for such tasks in the motion domain as coherent motion pattern filtering (rigid-body motion).

The application of this neural processing strategy to real-life problems is also illustrated. In particular, promising results have been obtained for an overtaking-car segmentation task. This problem is currently being addressed by many application-driven research groups [21–23]. In this scenario motion processing plays an important role, since an overtaking car exhibits a forward motion pattern in clear contrast to the overall backward motion pattern observed in the rear-view mirror due to the ego-motion of the host car.

2. Velocity Estimation Using a Neuronal Computation Scheme

The S&H model consists of two primary stages corresponding to cortical areas, the visual area 1 (V1) and the middle temporal area (MT), with parallel and regular computation in these layers (see Appendix A for a more detailed review of the model and configuration parameters).

The implementation of the S&H model, illustrated in Figure 1, for the application presented here can be summarized in five steps:

1. Compute local stimulus contrast.
2. Model simple and complex V1 neurons using space–time third Gaussian derivatives and spatial pooling (weighted output combination on a spatial neighbourhood region). This is done on three scales.
3. Combine the scales to adapt the computation to the rear-view-mirror perspective.
4. Model MT neurons summing the weighted responses of V1 cells that lie on its characteristic plane in the space–time domain.
5. Compute the velocity estimation for each pixel in the visual field using a weighted sum of winner neurons.

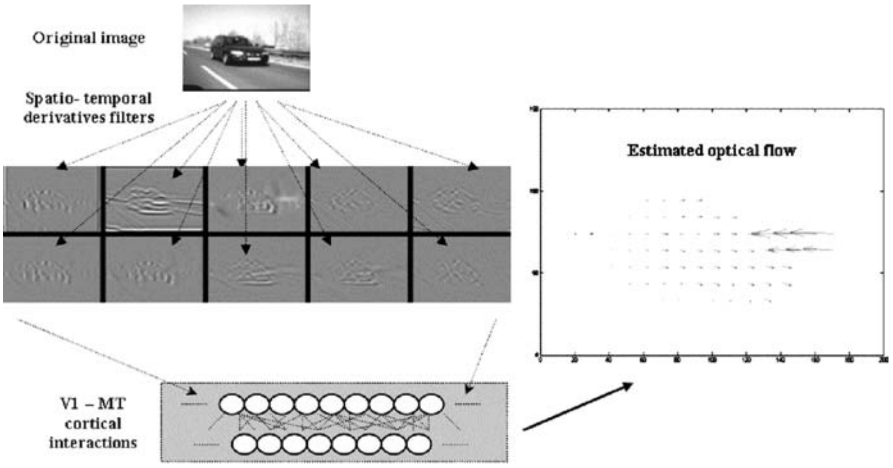
Original image
Spatio- temporal derivatives filters
Estimated optical flow
V1 – MT cortical interactions

Figure 1. S&H Model.

An overtaking-car sequence is used to evaluate the model. After convolution operations with Gaussian derivative filters, the pre-filtered images are combined to get V1 cell responses for different space–time orientations. These are combined to obtain the MT cell output. Finally, after a “some-winners-take-all” competition process only some neurons per pixel remain active. The initial velocity estimation is the weighted contribution of the active nodes whose inherent characteristics correspond to the estimated velocity vector.

For the overtaking-car application we adopt a special scale integration that takes into account perspective deformation to combine the different scales. The basic idea is to adopt a space-variant mapping strategy using small receptive fields on the left-hand side of the image (far visual field) and larger fields on the right-hand side (closer visual field). This works as an artificial fovea in the far visual field where high spatial resolution is desirable. The proposed model uses a “some-winners-take-all” configuration scheme among the MT population that selects only the MT cells with higher input, i.e. the ones that best match the local motion pattern, as shown in Figure 2.

The result of a plaid stimulus composed of one sinusoidal grating moving rightward and another moving downward forms a moving pattern toward the bottom-right corner (a). Gray levels represent a set of MT neuron responses. The relative position of the winner element with respect to the centre of the population represents the velocity module and direction. Maximum responses are given at the best-tuned MT neuron for that stimulus, but MT cells tuned to near velocities are not zero. We use a “some-winners-take-all” mechanism to estimate reliably the velocity in the presence of noise (b). Finally, the winner elements are those with responses close to the maximum response element (c).

3. Neural Collector Layer

Many studies suggest that the integration of local information permits the discrimination of objects in a noisy background [24–27]. The mechanism of this integration in biological systems is almost unknown, and although neurophysiological studies into primate visual cortex indicate that early visual processing occurs in

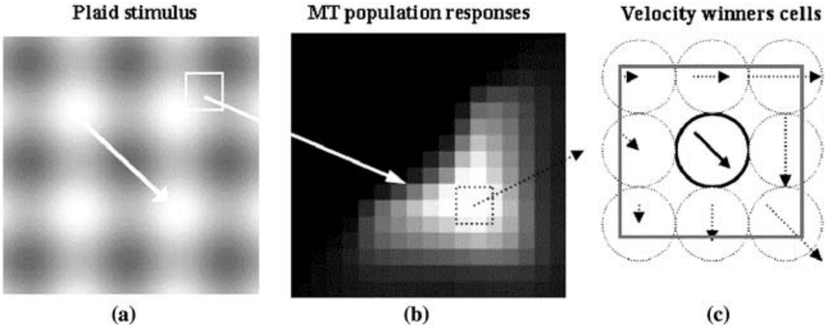

Figure 2. Example of population response.

a discrete manner, some evidence from single-cell recordings in monkey cortex suggests that local information is integrated into global patterns quite early on [27,28].

The new neural structure presented here can take advantage of population coding at the MT layer for a specific application such as the segmentation of overtaking cars. The main task of the proposed scheme is an improvement in the detection of rigid-body motion by the integration of local information about motion into global patterns. If we neglect possible rotations that are of only marginal importance for overtaking scenes, all points of a rigid body share the same speed and direction; isolated points inside a rigid body that move at other velocities are considered as noise.

In the proposed scheme, the MT layer is connected to this collector layer (CL). The cells at this stage receive excitatory, convergent, many-to-one connections from the MT layer and integrate the local information of motion into a region.

The CL has different collector neurons in the same area. Each CL cell is sensitive to a set of velocities $V \pm \Delta V$ from MT outputs, where ΔV represents slight variations in module and angle from preferred values, i.e. each CL neuron integrates the activity of 25 MT cells into a spatial neighbourhood that tunes the characteristic velocity of this CL neuron.

The CL is configured as a self-competitive layer: the collector neuron that receives the maximum contribution in its area of spatial influence inhibits the others and dominates (winner-takes-all). This function works as a filter that enhances the presence of rigid bodies and neglects noise patterns. Because the application addressed is focused on discriminating between leftward (ego-motion) and rightward (overtaking vehicle) moving features, only the cortical S&H neurons that match these directions are connected to the CL. The configuration of the CL neurons embodies another important aspect of the segmentation task: it can help to reduce the perspective deformations of motion patterns. Due to this effect, an overtaking vehicle, although moving at a constant speed, seems to accelerate as it approaches the host car, i.e. it moves more slowly when it is in the very left-hand side of the image (far away) and its speed increases when it moves rightward to a closer position. To reduce this effect, the distribution of each specialized collector neuron is non-uniform. The ratio of cells tuned to high speeds is lower on the left-hand side of the visual field than on the right-hand side. The opposite is done with cells more sensitive to slower speeds. This facilitates the detection of slow movements to the left of the visual field and rapid movements to the right and thus reduces the effects of perspective deformation.

Our recognition of the non-uniform distribution of cells throughout a neural layer is not a new concept; for instance the non-uniform distribution of cones and rods in the retina helps accurate sensing of diverse space–time patterns in different retinal areas [29]. The same perspective problem damages the perception of moving solid objects because the rear and front of the overtaking vehicle appear to be moving at slightly different speeds. This can be critical for very close vehicles. The

sensitivity of each CL cell to a set of characteristic speeds rather than a single one reduces the effect of this perception problem.
Furthermore, the winner neurons in any local influence area at the CL compete locally with other winner neurons from other areas in the neighborhood. This interaction facilitates the domination of large features and inhibits those winner neurons whose detected motion pattern is different with respect to the majority of the surrounding winner cells. Lateral inhibition is one of the most widely used mechanisms in nature; it has been used to explain the motion sensitivity of vision in frogs [30] and the low-luminance adaptation of cat retinal ganglion cells [31].

In this way, the output response of this filtering neural layer (CL) will be other than zero if there are winner collector neurons that are uninhibited by other winner cells (Figure 3). In Figure 3 C2 is inhibited by C3 because their selectivities lie in opposite directions, whereas C1 and C2 receive cross-excitation because their selectivity characteristics coincide (indicated by their inner arrows). This enhances coherent moving patterns and reduces fuzzy estimations.

Another property of CL neurons is a time constant which takes into account how the stimulus drives the onset and offset of the elements of this layer. If we make this time constant long, it means that more integration time from a lasting motion pattern is needed to activate a neuron and make it dominate against previously detected patterns. This also improves the stability of response for translational motion patterns in noisy environments and reduces velocity deformation due to perspective.

Using sparse motion estimation maps this strategy has revealed itself to be a robust scheme [32]. It detects areas within a population of features moving coherently and only patterns that activate a whole population of detectors with

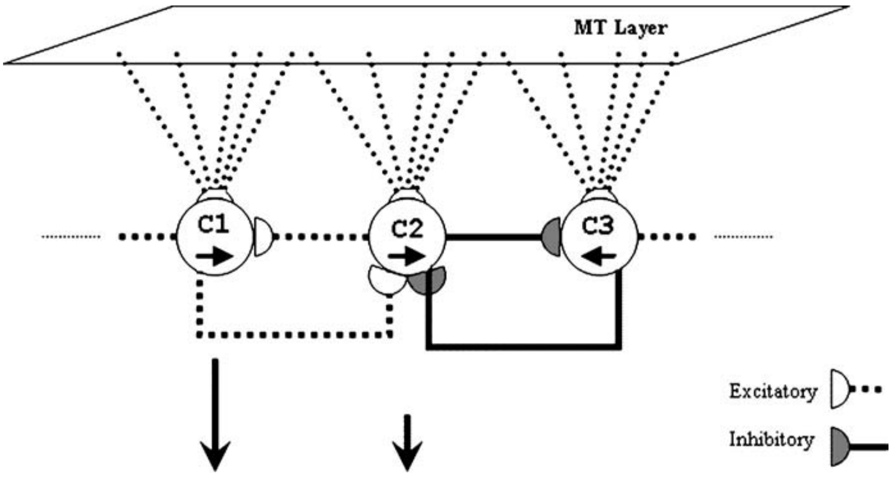

Figure 3. The figures shows the synaptic connections between three winner collector neurons that integrate the activity of MT cells of similar characteristics within a spatial neighbourhood. Two neurons detect rightward motion (→) and the third detects leftward motion (←).

a similar velocity pass through this filtering stage, becoming good candidates for a moving rigid body [32].

4. Experimental Results of the Overtaking-Car Segmentation Problem

To illustrate the processing scheme, the neural system described here has been applied to four real overtaking sequences. Figure 4 includes some illustrative results. In this figure, the overtaking vehicles have been surrounded with rectangular frames to facilitate the interpretation of the results. The proposed neural processing scheme efficiently segments rigid objects moving in opposite horizontal directions.

Figure 4a and b show an overtaking sequence with a dark car on a sunny day recorded with a conventional CCD camera. The other sequences were taken with a high-dynamic-range (HDR) camera. The sequences are: an overtaking truck (Figure 4c); a single overtaking-car on a foggy, rainy day (Figure 4d) and a sequence of multiple overtaking cars in a slight mist (Figure 4e).

Figure 4 is set out in columns. The left-hand column shows an original image of the overtaking sequences. The middle column shows the S&H extracted optical flow. The arrows show the motion direction (arrow sizes do not contain additional information due to the large range of velocities present in the sequences) and the grey scale indicates the speed (the lighter the grey the faster the motion). The right-hand column shows the CL outputs. The segmented overtaking car is represented as being dark in colour (rightward motion) and the background, moving in the opposite direction, is represented as being bright.

The receptive fields of the CL receive connections from MT neurons tuned to a cone of velocity directions focused mainly on horizontal motion. Thus the optical flow out of this cone is neglected by the CL, as can be seen in the bottom part of the car in Figure 4b. This figure also shows the effect of the spatio-temporal convolution operations of Simoncelli's model, which make the segmented overtaking car appear larger than in the original image. Nevertheless, since this effect is basically local spatio-temporal blurring, an object tracking system easily handles this local diffusion.

Some weather conditions (fog and rain) reduce the contrast in the sequences whilst car headlights would easily saturate CCD sensors and therefore open-air applications usually require HDR cameras. In spite of using these cameras, the extracted optical flow is worse in adverse weather conditions than in the sunny-day sequence, reducing the confidence of motion discrimination. Other effects that lead to worse car segmentation are reflections of light on the road and noisy artefacts produced by rain.

The HDR camera generates 10-bit precision whilst our model works at an 8-bit depth. This precision restriction induces other artefacts which lead to erroneous estimations of velocity that can be very significant in low-contrast sequences, as can be seen in Figure 4d. Nevertheless, the proposed neural computing scheme

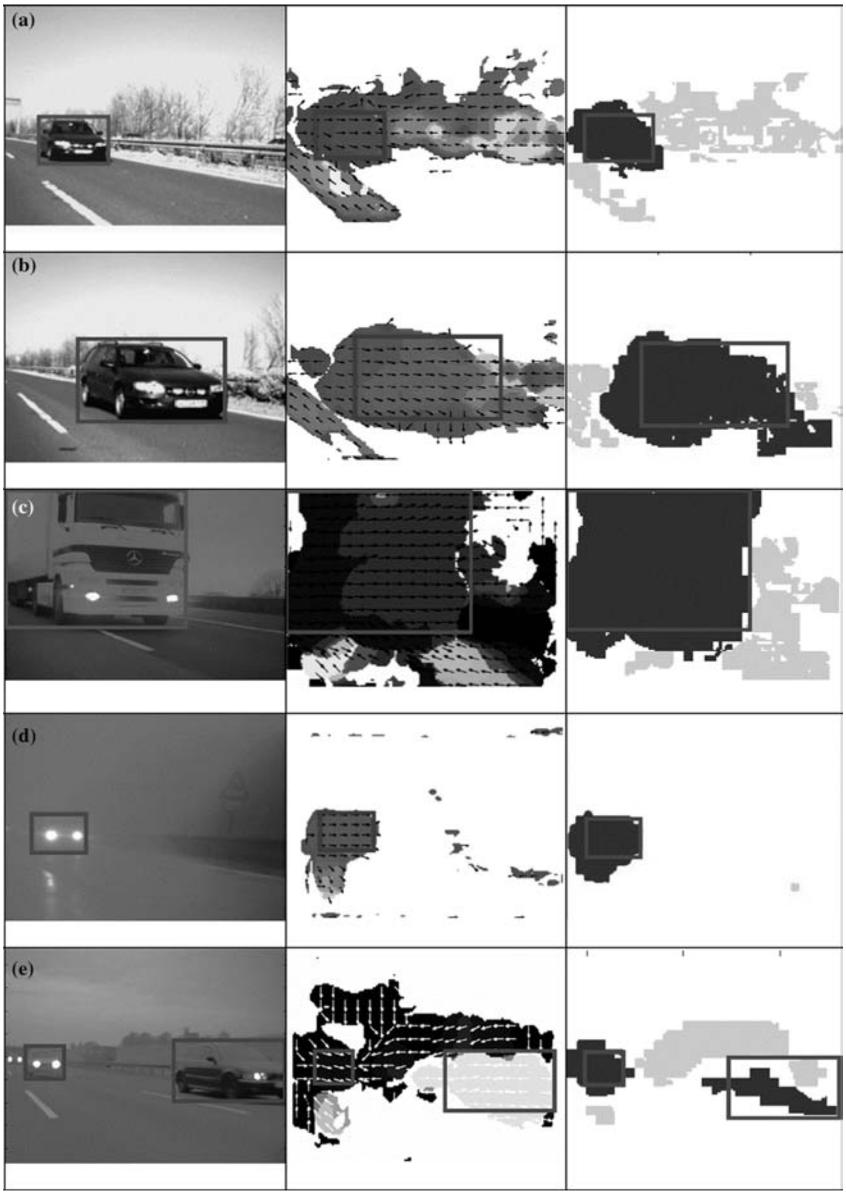

Figure 4. Overtaking-car sequence on a sunny day (a, b); on a cloudy day with slight mist (c, e); on a foggy, rainy day (d). The rectangles have been added manually to facilitate the performance evaluation procedure.

deals efficiently with all these artefacts in overtaking scenarios. On the right-hand column of Figure 4 it can be seen that the overtaking cars are accurately segmented from the background motion as homogeneous rightward-moving patterns.

The computational time of the processes is quite high. The motion-detection layer is the most computationally demanding stage (mainly performing

Table I. summarizes the segmentation performance with different car sizes and different visibility conditions.

		A	B	C	D	E
Nightfall	N_{fr}	35	41	39	40	1
	$N_{\mathrm{mr(in)}}$	200 ± 50	300 ± 100	700 ± 200	2100 ± 700	2629
	$N_{\mathrm{mr(out)}}$	100 ± 30	100 ± 50	40 ± 30	200 ± 100	217
	$\boldsymbol{P_{\mathrm{s}}}$	**0.70 ± 0.08**	**0.83 ± 0.10**	**0.95 ± 0.04**	**0.91 ± 0.03**	**0.92 ± 0.00**
Vehicle with mobile home	N_{fr}	16	23	23	15	24
	$N_{\mathrm{mr(in)}}$	200 ± 50	500 ± 100	1000 ± 300	2300 ± 500	7000 ± 1500
	$N_{\mathrm{mr(out)}}$	100 ± 50	100 ± 50	150 ± 70	100 ± 50	200 ± 100
	$\boldsymbol{P_{\mathrm{s}}}$	**0.72 ± 0.13**	**0.87 ± 0.07**	**0.88 ± 0.05**	**0.95 ± 0.01**	**0.97 ± 0.01**
Sunny day	N_{fr}	15	18	19	22	2
	$N_{\mathrm{mr(in)}}$	200 ± 20	300 ± 100	800 ± 300	2600 ± 1000	3000 ± 200
	$N_{\mathrm{mr(out)}}$	300 ± 20	200 ± 100	200 ± 100	900 ± 600	800 ± 10
	$\boldsymbol{P_{\mathrm{s}}}$	**0.44 ± 0.05**	**0.58 ± 0.15**	**0.78 ± 0.07**	**0.77 ± 0.06**	**0.80 ± 0.01**
Cloudy day	N_{fr}	19	29	25	27	13
	$N_{\mathrm{mr(in)}}$	100 ± 30	400 ± 100	900 ± 100	2700 ± 1500	4000 ± 800
	$N_{\mathrm{mr(out)}}$	4 ± 9	100 ± 50	100 ± 30	400 ± 300	400 ± 100
	$\boldsymbol{P_{\mathrm{s}}}$	**0.97 ± 0.07**	**0.85 ± 0.10**	**0.91 ± 0.02**	**0.88 ± 0.03**	**0.91 ± 0.01**

N_{fr} stands for the number of frames evaluated in each case. $N_{\mathrm{mr(in)}}$ and $N_{\mathrm{mr(out)}}$ are the average of rightward-moving features inside the rectangle (in) and outside (out), respectively, (both depend on the visibility of the sequence and the size of the car). P_{s} is the segmentation performance according to Equation (1).

spatio-temporal convolution operations). Convolutions are high time consuming operations on a generic processor. The population coding scheme also requires considerable memory resources. Despite this drawback, when computing the model on conventional processing platforms the system's architecture can be fairly well mapped on fine-grain parallel architectures such as FPGAs. Such hardware technology allows customized DSP processor design useful for embedded applications. Similar approaches have been implemented using this technology to achieve real-time processing [33, 34]. The hardware resources required for the motion layer can be reduced by implementing the Gaussian derivatives with recursive filters available in the literature [35], which reduce the external memory requirements. The parallelism of FPGA technology allows functional unit replication which can be used to implement multiple neuron processing in parallel.

The evaluation of a segmentation scheme with real images is not straightforward since the pixels are not labelled according to the object they belong to. To evaluate the performance of the described segmentation scheme in a overtaking-car scenario we have manually marked some overtaking sequences in different visibility conditions. The mark consists of a rectangle surrounding the overtaking car in

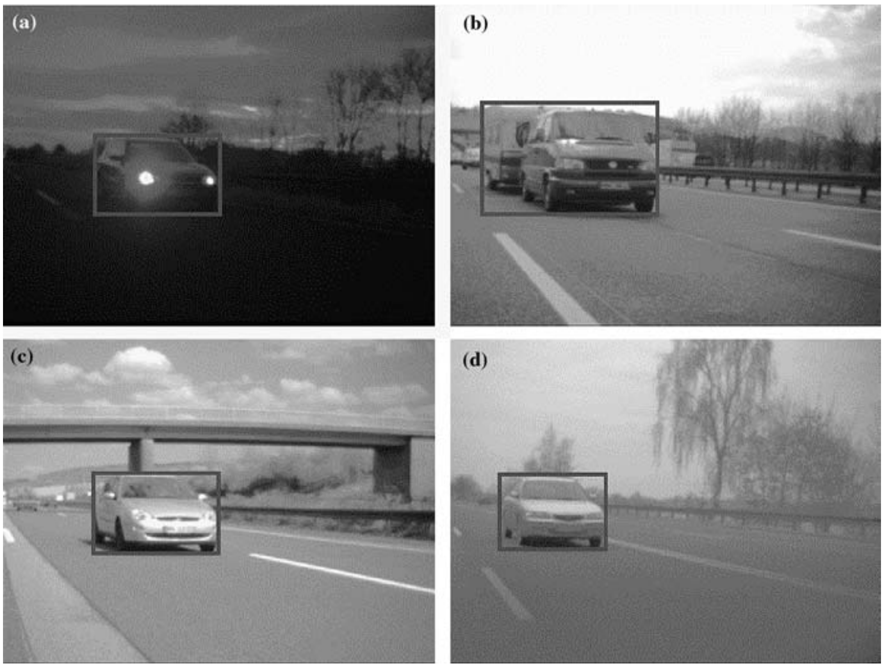

Figure 5. Four marked frames of the benchmark sequences: nightfall sequence (a), vehicle carrying a mobile home (b), sunny-day sequence (c) and cloudy-day sequence (d).

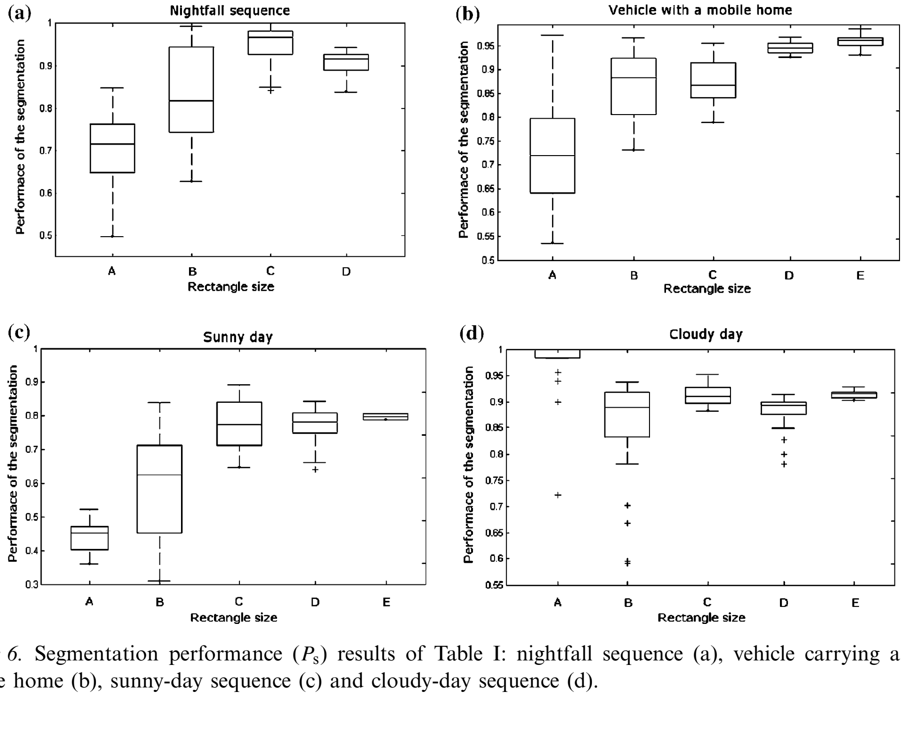

Figure 6. Segmentation performance (P_s) results of Table I: nightfall sequence (a), vehicle carrying a mobile home (b), sunny-day sequence (c) and cloudy-day sequence (d).

every frame. Table I summarizes the evaluation results from 446 benchmark frames from four overtaking sequences: sunny-day sequence, nightfall sequence (with the lights on), vehicle on a cloudy day with lights switched off and a vehicle carrying a mobile home. Four marked illustrative images of these sequences are included in Figure 5. Note that this benchmark set of frames does not correspond with any of the illustrative examples in Figure 4. In an overtaking scenario the motion structure of the approaching vehicle contrasts highly with the landmarks moving in the opposite direction due to the ego-motion of the host car. Therefore, in each image, we will label the motion features moving rightward as belonging to the overtaking vehicle and discard the rest. In a second stage we will calculate the performance of the segmentation (P_s) as the ratio between the features moving right inside the marked rectangle surrounding the car ($N_{\mathrm{mr(in)}}$ correctly labelled features) and all the features detected with rightward motion ($N_{\mathrm{mr(in)}} + N_{\mathrm{mr(out)}}$) according to Equation (1). The accuracy of the segmentation will depend on the relative speed and size of the overtaking car. We have distinguished different cases depending on the size of the car (due to the distance from the host vehicle). The car sizes in question, estimated in area (S) as the number of pixels contained in the rectangle, are: A ($315<S\leq 450$), B ($450<S\leq 875$), C ($875<S\leq 2220$), D ($2220<S\leq 6000$) and E ($S>6000$).

$$P_s = \frac{N_{\mathrm{mr(in)}}}{N_{\mathrm{mr(in)}} + N_{\mathrm{mr(out)}}}. \tag{1}$$

Figure 6 summarizes the results of Table I. It can be seen that performance increases as the overtaking vehicle approaches in all visibility conditions. The overtaking sequence on a sunny day occurs at a higher relative speed. Since the temporal convolution filters take into account 18 frames to estimate motion, the moving features detected appear to be delayed with respect to the overtaking car. This affects the performance results, which in the case of the sunny-day sequence are lower, as can be seen in Figure 6. Nevertheless, the overtaking vehicle is efficiently segmented and an object-tracking scheme could correct this artefact since the overall speed of the segmented object can be estimated as it evolves. Besides the results presented here, the proposed system has been tested with overtaking sequences at different relative speeds, from 10 to 20 m/s and reasonable performances were obtained. In Figure 5 we have included four frames taken from the benchmark sequences. The different contrast in the images depending on weather conditions is notable. Since all the parameters of the proposed scheme have been fixed for the whole evaluation process, this affects the motion estimation. In fact the sunny day sequence is composed of more highly contrasted images, which lead to more mismatches at the motion estimation stage than in the other cases. This could be corrected using larger spatial blurring filters in the pre-processing stage but that would affect the performance in other visibility conditions. Nevertheless, the performance results of Figure 6 are fairly stable in very different visibility conditions.

The results are very promising for this application. A simple object-tracking scheme based on the segmented features obtained would reliably track the overtaking car.

5. Conclusions

We describe a bio-inspired processing scheme for segmenting objects based on motion energy. A post-processing layer (the collector layer) filters the motion information from the MT layer. The topology of the CL connection embodies aspects that facilitate the segmentation of moving rigid bodies and thus reduces the effect of perspective deformation of the visual field due to the rear-view mirror.

The proposed neural system is highly parallel. It is a self-competitive neural computation scheme for feature selection. CL is a simple neural layer in which cells integrating motion information in a specific direction compete with other cells being activated by other motion directions. This competition inhibits local weak-motion patterns: only motion cues supported by neighbourhood cells remain active. This regularizes non-local but more complex motion patterns, therefore enhancing the capability of segmenting rigid bodies within noisy environments.

We report on how neural competitive mechanisms can be used in the framework of object segmentation based on motion cues. This has also been addressed recently by Fernández-Caballero *et al.* [36] with a multi-layer, motion-extraction neural network that includes learning and is able to detect efficiently both rigid and non-rigid objects. Contrary to this work, our contribution does not introduce a new motion extraction method but is based on an S&H approach. In fact, we propose a simple neural processing scheme for object segmentation (collector layer) that benefits from neural population coding produced by the bio-inspired motion-extraction strategy proposed by S&H. The proposed post-processing neural collector layer efficiently segments independent moving objects (IMOs) and regularizes noisy motion patterns.

Acknowledgments

This work has been supported by the European Project ECOVISION (IST-2001-32114) and the National Project DIPROVI (DPI2004-07032). We thank Hella (Dept. of predevelopment EE-11, Hella KG Hueck & Co., Germany) for providing us with the overtaking sequences and A. L. Tate for revising our English text.

Appendix A

A. NEURAL LAYER MODEL REVIEW AND CONFIGURATION PARAMETERS

The motion-estimation scheme described in the text is based on the MT model of Simoncelli [4, 6] with slight modifications motivated by computational aspects and velocity estimation accuracy. We use an interpolation function for the MT receptive field to maintain the directional selectivity of third Gaussian derivatives and Gaussian weighting coefficients for the pooling operation of V1 complex cells.

We also use a "some-winners-take-all" mechanism instead of a "winner-takes-all" to get continuous velocity estimation instead of a fixed set of output velocities (coded by single-cell responses).

The final system is the pyramidal structure shown in Figure 7, where each population of neurons tunes a different motion pattern.

A.1. V1 simple cells and complex cells

In the first layer of the S&H model, a linear model is used for V1 simple cells in the primary visual cortex, which exhibit specific selectivity for stimulus orientation and spatial frequency. A basic set of tuned V1 neurons covers a wide range of space–time frequencies with low overlapping. Each V1 neuron squares and normalizes its inputs. The next neural layer models V1 complex cells. They receive afferents from V1 simple cells distributed over a local spatial region, sharing the same space–time orientation and phase. V1 complex cells use spatial pooling, as suggested by Simoncelli *et al.* [4], (see Figure 8).

We model V1 cells in the primary visual cortex on the basis of their properties; they do not detect velocities but are rather directionally selective and tuned to space–time frequencies. According to this precept, a good function to model

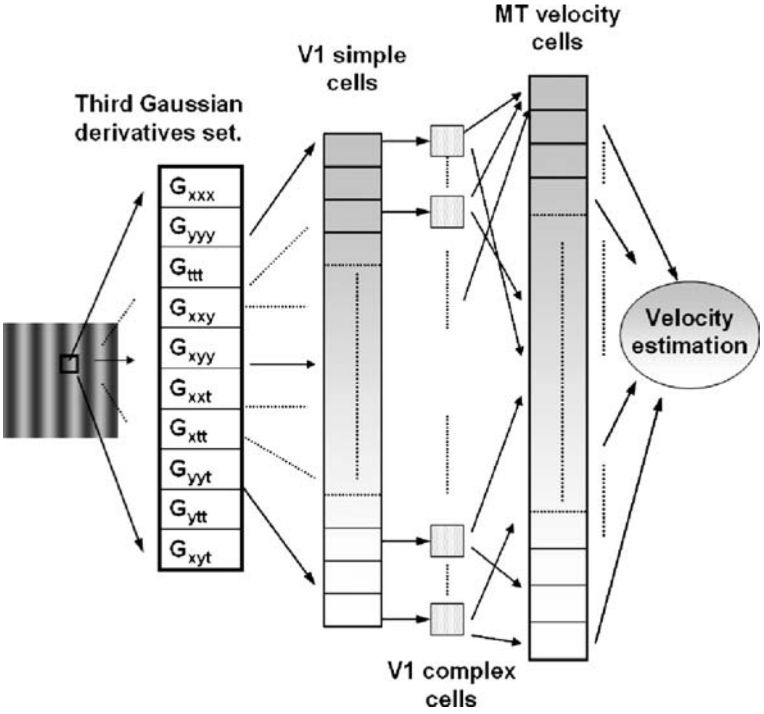

Figure 7. Pyramidal neuronal structure.

them is one of Gaussian derivatives because they are space–time directional, selective, band-pass filters. Gaussian derivatives are preferred to Gabor filters [37,38] because they are steerable [39] (see Appendix B, Section B.2) and therefore only 10 convolutions are needed to calculate the different space–time orientations. Section A.2 describes how to combine them to get local velocity selective cells and justifies the choice of using Gaussian kernels for the sake of computational efficiency.

We interpolate this set to form MT cell receptive fields. We employ a set of between 32 and 40 different space–time orientations, eight spatial and 4–5 temporal orientations. The receptive fields are computed on three different scales with variances of 1.4, 2.6 and 4.2 pixels. They are integrated into the MT stage. The response υ of a V1 simple cell on scale s is:

$$\boldsymbol{v_u^s} = \frac{\boldsymbol{K(DG3_u^s)^2}}{\sum\limits_{\boldsymbol{u}} \boldsymbol{(DG3_u^s)^2 + \sigma}}, \tag{2}$$

where $DG3_u^s$ is the third Gaussian derivative, with a particular orientation, u, on scale s, and σ and K are constants that determine the semi-saturation level and the maximum attainable response of the V1 simple neuron, respectively. Section B of this Appendix describes how to construct the directional derivatives and how to choose the appropriate variance values to tune the desired space–time frequency.

V1 complex cell responses are traditionally computed using energy models of motion with a pair of cuadrature filters (such as Gabor odd and even functions) with phases differing by 90°. Instead of combining over phase, equal results are found by spatial pooling, as shown in Figure 9.

The cells are modelled as Gabor filters (dashed line) and as squared Gaussian derivatives with spatial pooling (solid line). Similar results are found with both methods. The main difference is that non-zero response is obtained for bright and

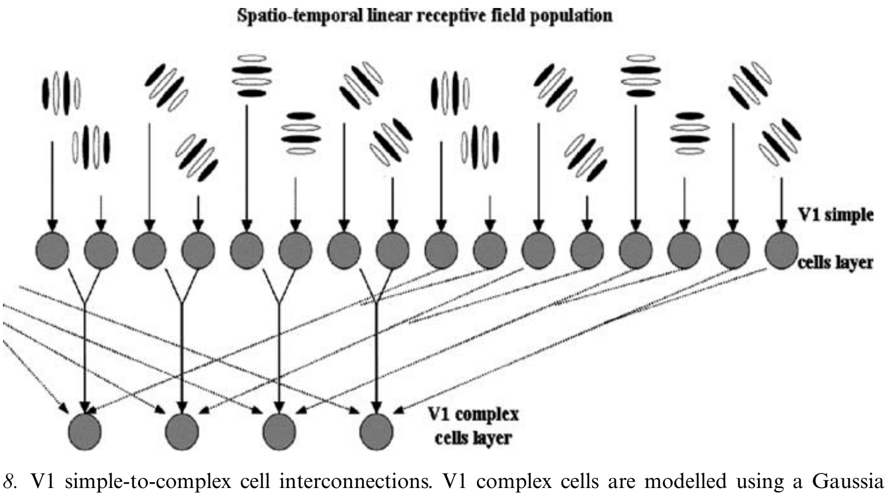

Figure 8. V1 simple-to-complex cell interconnections. V1 complex cells are modelled using a Gaussian pooling operation.

homogeneous areas although it is a flat area for these neurons' receptive fields. It can be seen (dashed line) on the left- and right-hand sides of the image for x <50 and x >90 that neuron response is not zero, unlike the Gaussian derivative filters. For this reason local contrast pre-filters are needed if Gabor-like neurons are used.

The V1 complex-cell response, at each scale C_u^s, is computed as the local average of simple cell responses with the same space–time orientation, u, and the same spatial scale, s:

$$C_u^s = \sum_{\Omega} w_{\Omega}^s v_u^s, \quad w_{\Omega}^s = Gaussian(x, y, \sigma') \tag{3}$$

where the pooling weights are spatial Gaussians with a variance, σ', slightly smaller than that of the V1 simple cells (typically $\sigma' = 0.9\sigma$) and truncated at 85% of their total power.

A.2. VELOCITY TUNING — MT CELL RECEPTIVE FIELDS

The MT cells are modelled by combining the outputs of a set of direction-selective V1 complex cells, the preferred space–time orientations of which are consistent with the MT cells' characteristic velocity. This combination depends on the first design stage where we select a velocity set that determines the tuning velocity of each MT cell. We have used a non-uniform distribution of 121 velocities that

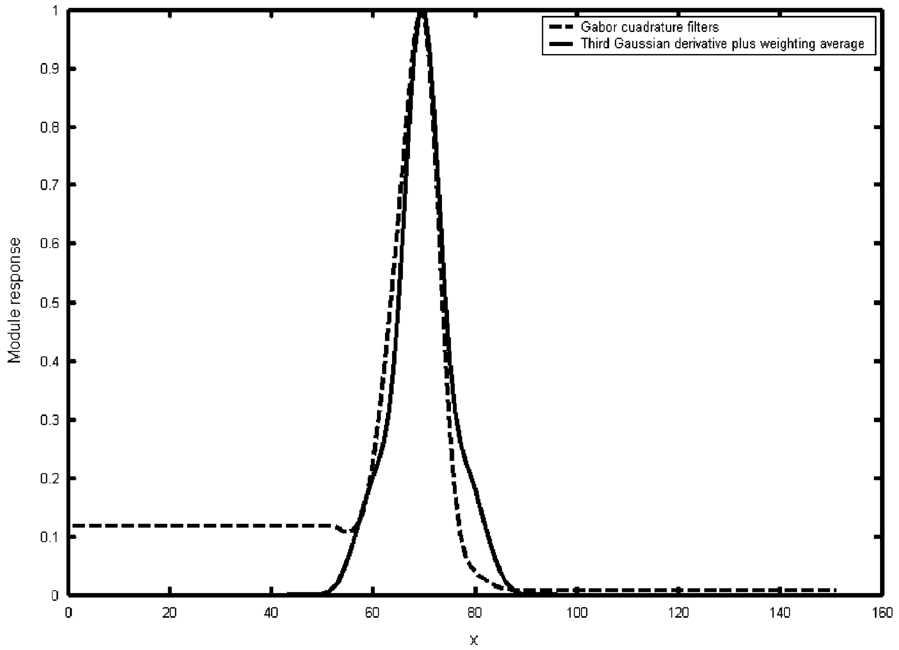

Figure 9. V1 complex-cell responses to a spatial abrupt image edge (bright area from x = 0 to x = 70 and dark area from this edge to the end).

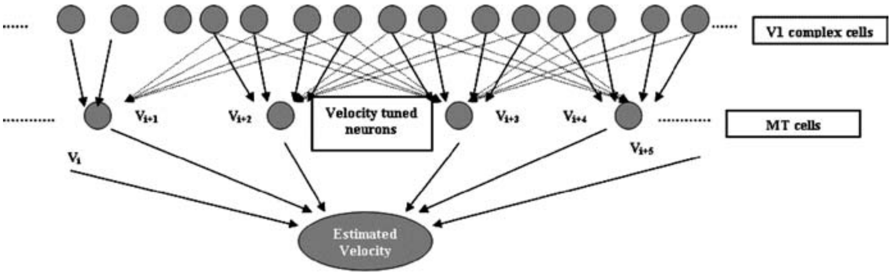

Figure 10. MT-V1 interactions. V1 complex cell, selective for space–time orientation on different scales, form the MT-cell receptive fields that allow velocity estimation.

cover the application requirements (in overtaking sequences, typically from −5 to 5 pixels/s).

The capacity of an MT cell for tuning a specific velocity can be easily understood in the frequency domain. In this domain the power spectrum of a translational pattern lies on the plane $(\omega_x, \omega_y) \cdot \vec{\mathbf{v}} + \omega_t = 0$ [40], and the tilt of the plane depends on the velocity. Each V1 cell has a spectral response that is crossed by infinite planes and therefore they are motion selective but not velocity selective. A set (at least 2) of this kind of cell is necessary to define unequivocally any particular velocity plane. Because we are using third-order directional Gaussian derivatives, which have a narrower orientation response, we need four or more V1 cells to cover the plane uniformly [4]. The intersection of constraints for MT cells tuned to the velocity $\vec{\mathbf{v}} = (\mathbf{v}_x, \mathbf{v}_y)$, as Simoncelli *et al.* [4] described, can be formed by V1 cells tuned to the space–time orientations:

$$\hat{\mathbf{v}}_1 = \begin{pmatrix} -\mathbf{v}_x \\ -\mathbf{v}_y \\ |\vec{\mathbf{v}}|^2 \end{pmatrix} \Big/ \sqrt{|\vec{\mathbf{v}}|^4 + |\vec{\mathbf{v}}|^2}, \quad \hat{u}_2 = \begin{pmatrix} -\mathbf{v}_\mathrm{y} \\ \mathbf{v}_x \\ 0 \end{pmatrix} \Big/ |\vec{\mathbf{v}}|,$$
$$\hat{u}_3 = (\hat{u}_1 + \hat{u}_2)/\sqrt{2}, \quad \hat{u}_4 = (\hat{u}_1 - \hat{u}_2)/\sqrt{2}. \tag{4}$$

A problem arises when an MT cell encodes a velocity and the V1 cell necessary to form its receptive field is not present in the V1 layer. In this case the receptive field of the MT cell interpolates between the available primitives. As mentioned in Section A.1, we only have 32–40 space–time orientations. Biological systems have a limited set of resources [41] and therefore not all the space–time orientations are covered by a specific cell.

Different combinations of V1 cells can be used to form the MT receptive fields. In our approach, we used a mechanism based on vector projection to obtain the interpolation weights. The idea is conceptually similar to that which Grzywacz & Yuille called "Ridge Strategy" [42] and to the least-squares fitting of Heeger *et al.* [3]. Excitatory connections from V1 cells to MT cells work as interpolation coefficients, with higher values for V1 tuned neurons closer to the desired orientation and pooling over scale (see Figure 10). We also use inhibitory connections for V1 cells tuned far away from the velocity plane. Note that these coefficients extend the interpolation properties of third Gaussian derivatives (which use a base of 10 elements) to the square derivatives. To make this interpolation properly an even larger set is needed, which is the reason for using 32–40 different V1 cells. Using this strategy, the MT cell receptive field is:

$$\boldsymbol{RF_{MT(v)}} = \sum_{s=1}^{3}\sum_{u}\sum_{i=1}^{4}(\boldsymbol{p}^{s}_{\boldsymbol{uiv}})^2\boldsymbol{C}^{s}_{\boldsymbol{u}}, \quad \boldsymbol{p_{uiv}} = (\hat{\boldsymbol{u}}_{\boldsymbol{iv}}\cdot\hat{\boldsymbol{u}}_{\boldsymbol{u}})^3, \tag{5}$$

where the sum s denotes weighting over scales, u the space–time orientation set (32 in our system) and i the interpolation operation over the orientations described in Equation (4). Note that we use positive and negative interpolation weights based on the angle between the desired and possible orientations of the V1 cell. The third power is selected to maintain the direction selectivity achieved using third Gaussian derivatives, and the tuning curves found with these interpolation weights produce narrow tuning responses.

An activation threshold is now used to inhibit responses lower than the mean value:

$$\boldsymbol{RF'_{MT(v)}} = \lfloor \boldsymbol{RF_{MT(v)}} - \boldsymbol{Mean}(\boldsymbol{RF_{MT(v)}})\rfloor, \tag{6}$$

where $\lfloor\rfloor$ indicates the rectification operation of the MT response. Equation (7) describes the MT cell responses:

$$\boldsymbol{MT_v} = \frac{\boldsymbol{K'}(\boldsymbol{RF'_{MT(v)}})^2}{\sum_{\boldsymbol{v}}(\boldsymbol{RF'_{MT(v)}})^2 + \boldsymbol{\sigma''}}, \tag{7}$$

where σ'' and K' are constants that determine the semi-saturation level and the maximum attainable response of the MT neuron, respectively.

One drawback of energy models is that they are contrast dependent [2]. We use a neuron model, Equation (7), that self-normalizes its activity [37], and with this and the competition layer scheme contrast dependency is reduced.

A.3. SCALE INTEGRATION

The scale integration that we introduce takes into account perspective deformation. The sum of the scales is computed using a combination of Gaussian weight functions throughout the image columns. The combination steps are illustrated in Figure 11 (the plots represent Equation (8)):

$$W_s = \exp\left[-(col-\mu_s)^2/(2*\sigma_s^2)\right] \quad V1_{eq} = \frac{\sum_s W_s * V1_s}{\sum_s W_s}, \tag{8}$$

where s indicates the scale, μ the spatial position on which this scale is centred, σ the scale variance of the Gaussian V1 receptive field and *col* position x throughout time.

Three scales of symmetrical receptive fields are used for the overtaking application with Gaussian weights. (a) Weights at different scales; (b) The equivalent filter scale as a function of position x in the visual field is shown in the plot on the right.

A.4. VELOCITY ESTIMATION

The strength of the MT responses allows us to determine the maximum probable velocity. Basically, the computational model can estimate velocity as it is coded in the winner MT cell (winner-takes-all mechanism). We, on the other hand, use a more relaxed "some-winners-take-all" scheme.

The velocity value is estimated by a quadratic interpolation of winner neurons. The responses above the dynamical threshold, given by $\kappa^*|MT_{VMAX}|$, produce winner neurons, where κ is the inhibition factor. The final estimated velocity is computed via Equation (9) with a typical inhibitory factor of $\kappa = 0.9$.

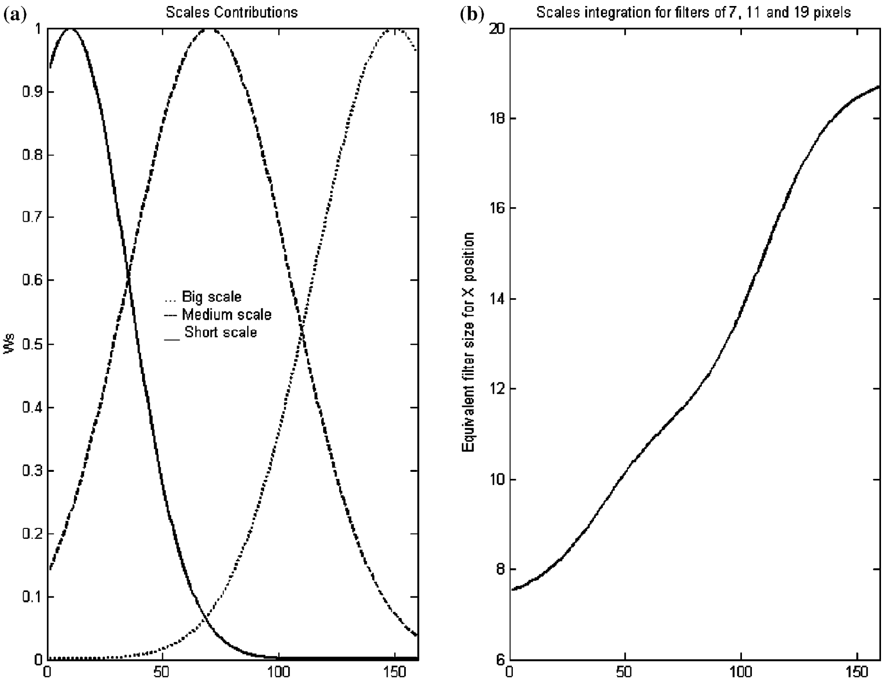

Figure 11. Integration of space–time scales. The plots on the left represent the weighted values of the contribution of the scales throughout the image (position x).

$$V = \sum_{v} \lambda \vec{v} \quad \lambda = \frac{\lfloor MT_v - \kappa |MT_v|_{MAX} \rfloor^2}{\sum_{v} \lfloor MT_v - \kappa |MT_v|_{MAX} \rfloor^2}. \tag{9}$$

One limitation for this neural structure is that it cannot detect second-order motion. Some modifications could be added to detect it [43] but this is outside the scope of our work here. Furthermore, the real-life application addressed here mainly requires accurate translational motion processing.

B. BASIC CONSIDERATIONS CONCERNING GAUSSIAN DERIVATIVES

Gaussian derivatives (as well as Gabor functions) are widely used as neuron receptive field models, as originally proposed by Young [44]. They have also been used to model the responses of disparity-sensitive neurons [45] or for the analysis of local orientation patterns in imagery [46]. The computational properties of Gaussians functions also make them very appropriate. Section B.1 describes the basic frequency tuning properties of Gaussian derivative kernels and Section B.2 shows the basic set used in three dimensions to make the directional derivatives.

B.1. 1-D GAUSSIAN DERIVATIVE KERNEL PROPERTIES

The well known equation of a 1-D Gaussian and its derivatives are:

$$g_0(x) = e^{\frac{x^2}{2\sigma^2}}, \quad g_n(x) = \frac{d^n}{dx^n} g_0(x) = P_{n,\sigma}(x) g_0(x). \tag{10}$$

Equation (10) indicates that the nth derivative of a Gaussian can be written as the product of a polynomial (generalized Hermite polynomial) by the original Gaussian. In the frequency domain Equation (10) can be expressed as

$$G_0(\omega) = \sigma e^{\frac{\sigma^2 \omega^2}{2}}, \quad G_n(\omega) = (j\omega)^n G_0(\omega) \tag{11}$$

The module response of the fifth Gaussian derivatives is shown in Figure 12. It should be noted that all Gaussian derivatives are band-pass with the only exception of the original Gaussian which is a low-pass filter. Furthermore, the Gaussian derivative bandwidths are approximately constant and asymptotically equal to: $\Delta\omega \to \frac{1}{\sqrt{2}\sigma}$ [47], with a preferred tuning frequency of $\omega_n = \sqrt{n/\sigma^2}$ [48]. These parameters allow us to choose the desired tuning frequency and bandwidth by selecting the derivative order and filter variance.

B.2. 3-D DIRECTIONAL GAUSSIAN DERIVATIVES

The V1 receptive fields used in the model are third-order Gaussian derivatives (GD3) along specific space–time directions. These have been used by different authors to model the early linear stages of the visual system [4,49]. The tuning (peak) frequencies of these filters are located over the surface of a sphere (more generally over an ellipsoid) for a given scale. These basic functions can be expressed as a linear combination of the 10 separable functions obtained by third-order derivation of a space–time Gaussian, g(x,y,t). The general expression of the separable basis of GD3 is the following in the space–time frequency domain:

$$\frac{\partial g_0(x,y,t)}{\partial x^{3-l-k}\partial y^l \partial t^k} \overset{F}{\longleftrightarrow} G_{3-l-k}(\omega_x,\omega_y,\omega_t) = (i\omega_x)^{3-l-k}(i\omega_y)^l(i\omega_t)^k G(\omega_x,\omega_y,\omega_t). \tag{12}$$

This filter set can be used to build third-order directional derivatives. One important property of Gaussian derivatives is that they are steerable filters [39]. This allows us to compute space–time oriented filters using a basic set of non-separable (N+1)(N+2)/2 filters, where N is the derivative order. In fact, Gaussian derivatives are (spherical) polar separable but not Cartesian separable. Thus it is computationally convenient to use these 10 separable filters, as indicated in Equation (12), as the basic set to compute oriented derivatives rather than use 10 non-Cartesian-separable oriented derivatives and steer between them. For a space–time orientation, the oriented derivative is computed as indicated in Equation (13):

$$D_u^3 g_0(x,y,t) = \sum_{l=0}^{3}\sum_{k=0}^{3-k}\left[\frac{3!}{l!k!(3-l-k)} u_x^l u_x^k u_x^{3-l-k} \frac{\partial g_0(x,y,t)}{\partial x^l \partial x^k \partial x^{3-l-k}}\right]. \tag{13}$$

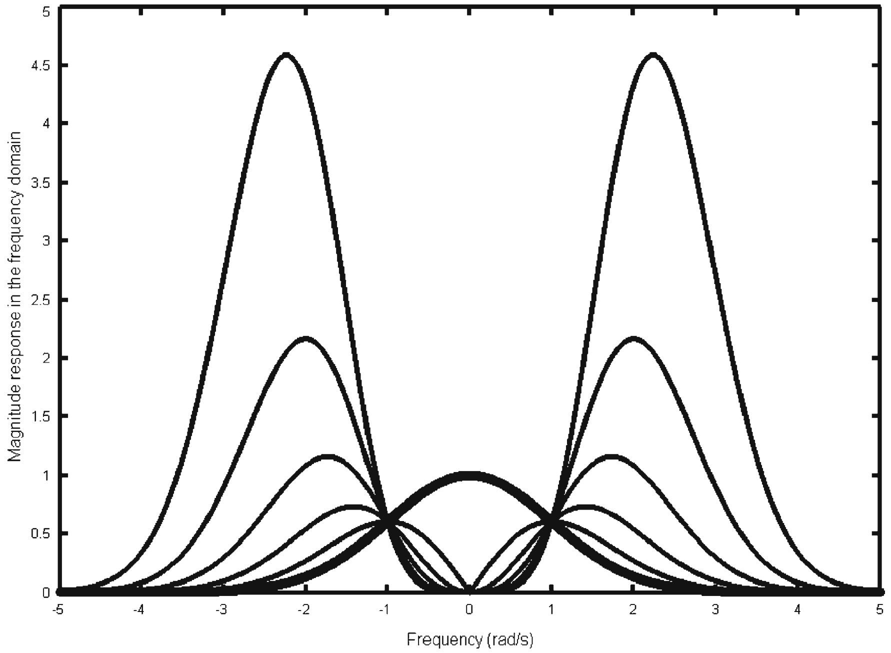

Figure 12. Gaussian 0 to 5 derivatives in the frequency domain.

The choice of a third-derivation order instead of a lower one, which would be easier to compute, is due to the high orientation discrimination desired. Space-time regions containing corners and transparencies or overlapping objects may have more than a single orientation at any given location. A filter such as GD1 or GD2 is unable to respond to the presence of two orientations at one point because of its limited angular resolution and therefore higher orders with a narrower frequency tuning are required [39]. Reasonable results can be obtained using the third-order derivative of a Gaussian, GD3. This approach allows the analysis of multiple oriented structures at a single point and the results when using this derivation order are biologically plausible [4].

References

1. Nakayama, K.: Biological image motion processing: a review, *Vision Research* **25**(5) (1985), 625–660.
2. Adelson, E. H. and Bergen, J. R.: The extraction of spatiotemporal energy in human and machine vision. In: *Proceeding of IEEE Workshop on Motion: Representation and Analysis*, Charleston, SC, pp. 151–156, 1986.
3. Heeger, D. J.: Model for the extraction of image flow, *Journal of the Optical Society of America* **4**(8) (1987), 1455–1471.
4. Simoncelli, E. P. and Heeger, D. J.: A model of neuronal responses in visual area MT., *Vision Research* **38**(5) (1998), 743–761.
5. Hubel, D. H. and Wiesel, T. N.: Receptive fields, binocular interactions and functional architecture in the cat's visual cortex. *Journal of Physiology* **160** (1962), 106–154.
6. Simoncelli, E. P.: *Distributed analysis and representation of visual motion, PhD thesis*, Massachusetts Institute of Technology, Deptartment of Electrical Engineering Computer Science, Cambridge, MA., 1993.
7. Bors, A. G. and Pitas, I.: Moving scene segmentation using median radial basis function network. *In Proceedings of IEEE Symposium on Circuits and Signals (ISCAS'97)* Vol. I, pp. 529–532, Hong Kong, 1997.
8. Chen, Y. -K. and Kung, S. Y.: A multi-module minimization neural network for motion-based scene segmentation. *In Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, pp. 371–380, Kyoto, Japan, 1996.
9. Shi, B. E. and Boahen, K. A.: Competitively coupled orientation selective cellular neural networks, *IEEE Transactions on Circuits and Systems I* **49**(3) (2002), 388–394.
10. Ros, E., Pelayo, F. J., Palomar, D., Rojas, I., Bernier, J. L. and Prieto, A.: Stimulus correlation and adaptive local motion detection using spiking neurons, *International Journal of Neural Systems* **9**(5) (1999), 485–490.
11. Harrison, R. R. and Koch, C.: A robust analog VLSI Reichardt motion sensor, *Analog Integrated Circuits and Signal Processing*, **24** (2000), 213–229.
12. Andreou, A. G. and Strohbehn, K.: Analog implementation of the Hassenstein-Reichardt-Poggio models for vision computation. *In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp. 707–710, 1990.
13. Stocker, A. A.: Analog VLSI focal-plane array with dynamic connections for the estimation of piecewise-smooth optical flow, *IEEE Transactions on Circuits and Systems-1: Special Issue on CNN Technology and Active Wave Computing*. **51**(5) (2004), 963–973.
14. Kramer, J., Sarpeshkar R. and Koch C.: Pulse-based analog VLSI velocity sensors, *IEEE Transaction Circuits and System II: Analog and Digital Signal Processing* **44**(2) (1997), 86–101.

15. Indiveri, G.: Smart adaptive systems on silicon, In: M. Valle (ed.), *Neuromorphic Engineering*, Kluwer Academic Publishers: Boston, MA 2004.
16. Etienne-Cummings, R.: Intelligent robot vision sensors in VLSI, Autonomous Robots **7** (1999), 225–237.
17. Stocker, A. and Simoncelli, E.: Constraining a bayesian model of human visual speed perception. *In Proceedings of NIPS Neural Information Processing Systems* 17, Vancouver, Canada, 2004.
18. Barron, J., Fleet, D. and Beauchemin, S.: Performance of optical flow techniques, *International Journal of Computer Vision* **12**(1) (1994), 43–77.
19. Liu, H., Hong, T. H., Herman, M., Camus, T. and Chellappa, R.: Accuracy vs. efficiency trade-offs in optical flow algorithms, *Computer Vision and Image Understanding* **72**(3) (1998), 271–286.
20. McCane, B., Novins, K., Crannitch, D. and Galvin, B.: On benchmarking optical flow, *Computer Vision and Image Understanding* **84** (2001), 126–143.
21. Franke, U., Gavrila, D., Gern, A., Görzig, S., Janssen, R., Paetzold, F. and Wöhler, C.: From door to door- principles and application on computer vision for driver assistant systems, In: L Vlasic, F. Harashima, and M. Parent (eds.), *Intelligent Vehicle Technologies: Theory and Applications*, pp. 131–188, Butterworth: London, UK. 2000.
22. Handmann, U., Kalinke, T., Tzomakas, C., Werner, M. and von Seelen, W.: Computer vision for driver assistance systems, *In Proceedings of SPIE* Vol. 3364 pp. 136–147, Orlando, 1998.
23. Görzig, S. and Franke, U.: ANTS-Intelligent vision in urban traffic, in *IEEE Conference on Intelligent Transportation Systems*, Stuttgart 1998.
24. Barlow, H. B.: The efficiency of detecting changes of intensity in random dot patterns, *Vision Research* **18**(6) (1978), 637–650.
25. Field, D. J., Hayes, A. and Hess, R. F.: Contour integration by the human visual system: evidence for local "association field", *Vision Research* **33**(2) (1993), 173–193.
26. Saarinen, J., Levi, D. M and Shen, B.: Integration of local pattern elements into a global shape in human vision, *In Proceeding of the National Academic of Sciences USA*, **94**, pp. 8267–8271, 1997.
27. Gilbert, C. D. and Wiesel, T. N.: Intrinsic connectivity and receptive field properties in visual cortex, *Vision Research* **25**(3) (1985), 365–374.
28. Grosof, D. H., Shapley, R. M. and Hawken, M. J.: Macaque V1 neurons can signal 'illusory' contours. *Nature*, **365**(1993), 550–552.
29. Hubel, D. H.: *Eye, Brain and Vision*, Scientific American Library: New York, 1988.
30. Lettvin, J. Y., Maturana, H. R., McCulloch, W. S. and Pitts, W. H.: What the frog's eye tells the frog's brain, in *Proceedings of IRE*, 47, pp. 1940–1951, 1959.
31. Enroth-Cugell, C. and Robson, J. C.: The contrast sensitivity of retinal ganglion cells of the cat. *Journal of Physiology*, **187** (1966), 517–552.
32. Mota, S., Ros, E., Díaz, J., Botella, G.,Vargas, F. and Prieto, A.: Motion driven segmentation scheme for car overtaking sequences. *In Proceedings of 10th International Conference on Vision in Vehicles (VIV'2003)*, Granada, Spain, 2003.
33. Díaz, J., Ros, E., Mota, S., Carrillo R. and Agís R.: Real time optical flow processing system, *Lecture Notes in Computer Science* **3203** (2004), 617–626.
34. Mota, S., Ros, E., Díaz, J., Ortigosa, E. M., Agís, R. and Carrillo, R.: Real-time visual motion detection of overtaking cars for driving assistance using FPGAs, *Lecture Notes in Computer Science* **3203** (2004), 1158–1161.
35. van Vliet, L. J., Young, I. T. and Verbeek, P. W.: Recursive gaussian derivative filters. *In Proceedings of the 14th International Conference on Pattern Recognition*, ICPR'98, 509–514, Brisbane, Australia, 1998.

36. Fernández-Caballero, A., Mira, J., Fernández, M. A., Delgado, A. E. On motion detection through a multi-layer neural network architecture, *Neural Network* **16** (2003), 205–222.
37. Heeger, D. J.: Normalization of cell responses in cat striate cortex, *Visual Neuroscience*, **9**(2) (1992), 181–198.
38. Fleet, D. J., Wagner, H. and Heeger, D. J.: Neural encoding of binocular disparity: energy models, position shifts and phase shifts, *Vision Research*, **36**(12) (1996), 1839–1857.
39. Freeman, W. T. and Adelson, E. H.: The design and use of steerable filters, *IEEE Pattern Analysis and Machine Intelligence*, **13**(9) (1991), 891–906.
40. Watson, A. B. and Ahumada, A. J.: A look at motion in the frequency domain, In: Tsosos, J.K. (ed.), *Motion: Perception and representation*, pp. 1–10. New York, 1983.
41. Nauta, W. J. H., Freitag, M.: *Fundamental Neuroanatomy*, W.H. Freeman, New York, 1986.
42. Grzywacz, N. M. and Yuille, A. L.: A model for the estimate of local velocity by cells in the visual cortex, *Proceedings of the Royal Society of London B* **239**, pp. 129–161, 1990.
43. Sperling, G., Chubb, C., Solomon, J. A. and Lu, Z. -L.: Full-wave and half-wave processes in second order motion and texture, *in* Wiley (Ciba Foundation Symposium, 184) *Higher-order processing in the visual system*, pp. 287–303, Chichester: U.K, 1994.
44. Young, R. A.: Simulation of human retinal function with the Gaussian derivative model, *in Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* pp. 564–569, Miami, FL, 1986.
45. Frye, R. E. and Ledley, R. S.: Derivative of Gaussian functions as receptive field models for disparity sensitive neurons of the visual cortex, *Proceedings of the 1996 Fifteenth Southern Biomedical Engineering Conference*, pp. 270–273, 1996.
46. Simoncelli, E. P., Farid, H.: Steerable wedge filters for local orientation analysis, *IEEE Trans Image Proceedings* **5**(9) (1996), 1377–1382.
47. Koenderink, J. J. and van Doom, A. J.: Representation of local geometry in the visual system, *Biological Cybernetics*, **55**(6) (1987), 367–375.
48. Bloom, J. A. and Reed, T. R.: A Gaussian derivative-based transform, *IEEE Transactions on Image Processing*, **5**(3) (1996), 551–553.
49. Young, R. A. and Lesperance, R. M., A physiological model of motion analysis for machine vision, in *Proc. of the SPIE*, 1913 pp. 48–123, San Jose, CA, 1993.