ELSEVIER

Image and Vision Computing 16 (1998) 89–98

image AND vision COMPUTING

Proper scale for modeling visual data

Franc Solina*, Aleš Leonardis

Computer Vision Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Tržaška 25, SI-1001 Ljubljana, Slovenia

Received 4 June 1996; accepted 4 June 1997

Abstract

We propose a method for determining the proper scale for modeling visual data. An efficient architecture for selective image modeling is discussed which selects models according to the task, the nature of the scene and the computational constraints. We give an example in which models of different scales are recovered in parallel and show that this redundant representation can effectively be pruned using the criterion of *Minimal Description Length*. Models that are selected in the final description indicate the appropriate scale of observation. © 1998 Elsevier Science B.V.

Keywords: Scale; Image modeling; Vision architecture

1. Introduction

It is becoming clear in the vision community that a single, universal, all-purposeful representation is not feasible. Such a representation would have to be excessively flexible with many parameters and hence computationally unstable. Instead of one complex model, several elementary models combined can account for the same phenomena which is amply demonstrated by biological visual systems and experiments. Instead of a universal representation for computer vision we should try to find a robust and universal computer vision architecture that would consist of several modules, that could combine different sources of information and readily adapt itself to specific goals. These ideas are known in literature as *active* [1] and *purposive* [2] vision. Hence, in a general purpose vision system, a thorough reconstruction of shape and depth from different visual cues is neither necessary nor sensible, and maybe not even possible to accomplish.

But, the question is which models to use and where in the scene to apply them in a particular case? Mobile robots, for example, that have to perform in a complex environment must concentrate only on those features of the environment that are relevant for their goal and cannot waste their resources. If a wheeled mobile robot has to move in a specific direction, its vision system must find flat surfaces, free of obstacles, on which it can drive forward. Appropriate, 'natural' models for such a goal could be, for example, planar patches for the driving surface and volumetric models for obstacles. The scale of those models must take into account the size of the vehicle (i.e. radius of wheels) and ignore smaller features (i.e. stones on a gravel road). If, on the other hand, the vision system must guide grasping, volumetric models of scale reflecting the construction and size of the mechanical hand would probably be a better choice rather than a large set of surface patches. Such task-dependent modeling also simplifies further reasoning about a particular task. The difficult problem of functionality in vision [3] can be approached through selective reconstruction of specific models tailored to particular functions, such as, driving, grasping, sitting, etc. In contrast, we believe that finding some particular functional features in a complete reconstruction is much more difficult.

The nature of the scene sets additional constraints on the selection of models. To represent different image qualities different models are needed (for shape, texture, color, specularity, etc.). However, depending on the required scale of modeling, the appropriate type of models for a particular part of the image might change. When fine grained modeling is called for, a particular texture model would have to be applied whereas a simple surface model would suffice on a coarser scale.

Another strong reason for selective scene modeling is processing constraints. If a vision system is to perform in real situations it must devote all of its computing resources first to those parts of the scene that are most relevant for accomplishing the goals. These image parts are in turn determined by using task-oriented models which find their own domain of applicability in the image.

In summary, selective reconstruction or modeling of the

* Corresponding author. E-mail: franc.solina@fri.uni-lj.si.

PII S0262-8856(97)00052-8

scene should be made such that:

- models tailored to the task are used;
- the image itself determines the applicability of the models; and
- resource and time constraints are taken into consideration.

In addition, models and their recovery methods must also provide constraints (additional information) needed to cope with problems of insufficient and unreliable information.

In a sense, application oriented vision systems used this reasoning all the time by hardcoding particular models of certain scale appropriate for its application. Research, on the other hand, has for quite some time been oriented towards finding a universal representation that would support all kinds of visual tasks [4]. It is clear now that the search for a single, universal all-purposeful representation is misguided. Thus, a robust universal architecture that can adapt itself to different tasks and combine several different models is needed.

1.1. Universal computer vision architecture

We propose a universal visual architecture (Fig. 1) which can perform selective recovery as described above. Tasks defined by an outside agent determine various potential supportive models together with the appropriate image domain for each model type. In the driving example, the system would search for planar driving surfaces in the lower half of the image. For grasping, the system would search for graspable objects of appropriate size and shape within reach of the system. In case of image sequences, the results of processing the previous image frame could be used to instantiate proper models in the corresponding image domain. At instantiation, the system can search for instances of all model types in the whole image.

Several models of either parametric [5,6] or rigid [7] nature can be chosen and searched for in parallel. Unlike 'classical' segmentation that attempts to describe the whole image with a particular type of model, this scheme recovers only those models that are applicable for a specific task at hand. Parts of the image where there is not enough local support for a model (i.e. no model is applicable) remain undescribed. Apparently, those parts of the image are not important for accomplishing the task of the system. On the other hand, some parts in the image may be, at least partially, matched by multiple models, resulting in a redundant description of the image. Thus, to get concise description information from all recovered models, an efficient selection procedure has to be designed.

1.2. Model selection

In [8,5] we described a procedure for selecting amongst competing models of the same kind concurrently with the recovery process where the only difference between the models was their initial spatial position in an image. Examples of this procedure applied to segmentation of range images are shown later. In this paper we study the

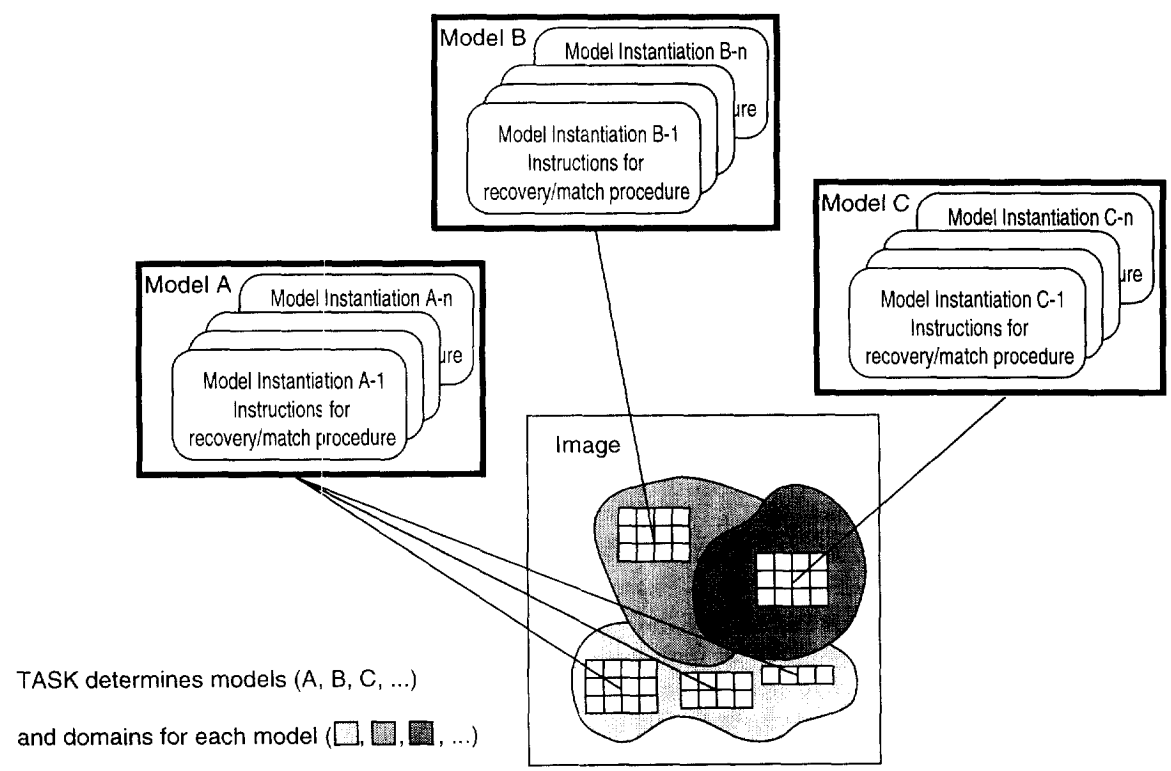

Fig. 1. Architecture for selective scene modeling. Each task (defined by an outside agent) determines a set of preferred models. Several instances of each type of model start detection in their corresponding seed regions. Domains (whole/part image) for each model type are defined by the task.

problem of selecting models of appropriate scale among possible different types of models.

The criteria for selection should satisfy, besides specific task-dependent constraints, a more general principle, for example, Minimum Description Length. Leclerk [9] wrote:

> Designing a language for highly textured scenes, and combining this with the language for smooth scenes in such a way that the more efficient language is chosen for different parts is extremely challenging.

In Leclerc's case, the models (language) incorporated in the objective function, which is designed to estimate models based on the MDL principle, have a strong impact on the computational feasibility of the procedure. Thus, the language has to be designed so as to facilitate the optimization procedure. In the paradigm presented in this article, on the other hand, model-recovery and model-selection are two independent processes, the selection procedure is independent of a particular type of models (language). This enables the interaction amongst models of completely different types.

In general, models can possess a variety of mutually exclusive or inclusive pieces of knowledge. We propose here a way of combining multiple sources of information through Minimum Description Length (or a similar measure) with special emphasis on how this principle automatically determines the scale on which phenomena should be observed. Models differ in the amount of allowable deviation from the model, which results in a set of interpretations that encompass different scales. This notion of multiple scale modeling is somewhat different from the standard multiresolution which is normally based on filtering the original image with a set of different spatial operators to obtain a hierarchy of images of different resolutions. Here, the original image remains intact, only the extent and allowable deviations of models are changed.

In the next section, we briefly present a procedure for obtaining potential descriptions of a scene from several competing model recovery processes. In the following section we define the use of the MDL principle for combining multiple sources of information and show a simple example that demonstrates how both the type of the models and the design of a criterion function determine the outcome of the selection procedure. At the end of the paper some actual models are recovered from real images in order to analyze and compare them using the described MDL principle.

2. Multiple sources of information

The term multiple sources of information has to be understood in the sense that each model represents a different feature which is present in an image and thus representing a separate source of information.

We begin with a brief outline of the model detection and/or model recovery process. The image plane on which the search for models is to be performed is denoted by I. The input image data is given by a (possibly vector-valued) function $g(x)$ defined on domain $\mathcal{E}$ where $\mathcal{E} \subseteq I$.

The general problem can be stated as follows:

> Given the set of all image points on the domain $\mathcal{E}$ and a logical uniformity predicate $P(\cdot)$ which determines the conformity between the data and the models, find a set of parametric models $M = \{M_1, ..., M_{n_R}\}$ and a set of corresponding domains (regions) $R = \{R_1, ..., R_{n_R}\}$, $\forall i : R_i \subseteq \mathcal{E}$. n_R is the number of such regions found in an image.

In general, the following conditions hold

$$\bigcup_{i=1}^{n_R} R_i \neq \mathcal{E}. \tag{1}$$

It states that not *all* data points in $\mathcal{E}$ are necessarily described by the models, which effectively means that the models define their own *domain of applicability* within $\mathcal{E}$. It is also possible that some of the data points simultaneously belong to more than one model.

We have explicitly stated these two conditions in order to contrast the approach used in this article which is essentially a search for a set of pre-defined models (parametric, non-parametric) to the classical segmentation problem defined by [10], for which the following two conditions hold for the set of models:

$$\bigcup_{i=1}^{n_R} R_i = \mathcal{E} \tag{2}$$

$$R_i \cap R_j = \varnothing \quad \forall i \neq j \tag{3}$$

However, the additional two conditions concerning the uniformity predicate are valid in both cases:

$$\forall i : P(R_i) = \textbf{TRUE} \tag{4}$$

and if R_i is adjacent to R_j

$$P(R_i \cup R_j) = \textbf{FALSE} \; i \neq j. \tag{5}$$

The decision to formulate our approach as a search problem has many important implications:

- Results will explain only those parts of an image that, in terms of a goodness-of-fit measure, have enough resemblance to the primitives. In other words, the method automatically determines its domain of applicability.
- The independent search will produce several possible interpretations which will compete with each other to form the best description of the image.
- Search performed independently by different modules can be easily parallelized.

The final outcome of the model recovery procedure for a model m_i consists of three terms:

1. The region R_i, which represents the domain of the model

and encompasses $n_i = |R_i|$ image elements that belong to the model.
2. The set of model parameters $\boldsymbol{a}_i$ (N_i denotes the cardinality of this set).
3. The error-of-fit measure ξ_i which evaluates the conformity between the data and the model.

While this description is general, i.e. independent of a particular choice of models, specific procedures designed to operate on individual type of models can differ significantly. This is primarily due to the matching and fitting processes which depend on the type of models and the choice in defining the measure of the distance between the model and the data.

3. Selection of models operating on multiple scales

Consider examples shown in Fig. 2. Which is the number of dots in a line when the perception of individual dots switches to the perception of a line (a)? How strong must the directional discontinuity of connected straight line segments be to change the perception of a single line to several individual line segments (b)? When does the explicit perception of individual squares in a checkerboard pattern change to the perception of a single square of particular texture (c)?

The importance of various representations (models) of an object has been emphasized by Bobick and Bolles [11], however the *prescribed* decision mechanism that is responsible for switching between different representations is based on an absolute criterion, namely, only when a currently used model fails, is a different model tested. Besides, it does not take into account the complexity of individual descriptions. On the other hand, in the paradigm outlined here, multiple representations are compared on a relative basis—they are generated (thus they exist) simultaneously, and the decision as to which of these is more appropriate (depending on certain criteria) is based on their relative comparison. Moreover, the complexity of individual descriptions is an explicit feature of our approach.

We put the task of combining and selecting among

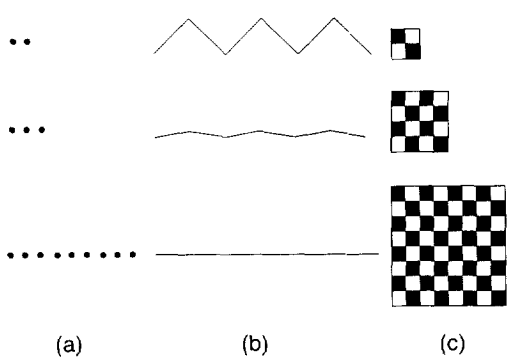

Fig. 2. When does the representation switch from modeling single elements to modeling the overall structure? Examples: (a) individual dots vs a line; (b) multiple line segments vs a single line; (c) individual squares vs a textured plane.

different models in the framework of a selection procedure where recovered individual descriptions compete as hypotheses to be accepted in the final interpretation of the data. The selection procedure, which is thus performed on the level of primitives rather than on the level of their constituent elements, i.e. pixels, is reduced to the optimization of an objective function which has a general form:

$$F(m) = \sum_{i}^{M} m_i \mathcal{L}_{m_i}, \quad m = [m_1, ..., m_2, ..., m_M]^T \tag{6}$$

where m_i denotes a *presence variable* having the value 1 for the presence of the particular model and 0 for its absence in the final description. $\mathcal{L}_{m_i}$ denotes the weight of a particular model.

The difficulty of the problem is in how to ensure that the selected set of models will be optimal. Since the requirements of the task have already been considered in the choice of the types of the models, we believe that there are mechanisms that operate on a more general level in order to reduce the number of redundant descriptions that are subsequently passed to other modules for further processing. There are computational constraints on the amount of information that can be computed in finite time and with finite memory for further processing (bounded rationality—Simon [12]). Intuitively, this reduction in complexity of a representation coincides with a general notion of simplicity, which has a long history. Gestalt psychologists summarized their observations in a number of Gestalt principles, one of them being the law of Prägnanz, or the minimum principle, which states that the visual field will be organized in the *simplest* or *the most likely* possible way [13]. In information science, Shannon [14] revealed the importance of the relation between the probability theory and the shortest encoding (simplicity). Recently, simplicity in terms of the MDL principle, defined by Rissanen [15–17], has found its applications in computer vision [7,9,18,19].

Thus, our goal is to select a description that minimizes Eq. (6) under the condition that portions of the image that can be described (at least one of the models is applicable) must be described (with at least one model). The weight $\mathcal{L}_{m_i}$ is proportional to the descriptional complexity of a particular model and can be modified according to the interactions between the models.

3.1. MDL principle

The total number of bits L required to encode the observed data is the sum of bits $L_{D|M}$ required to encode the data D, given the model M, and the number of bits L_M required to encode the parameters of the model M

$$L = L_{D|M} + L_M \tag{7}$$

The measure $-\log P(x|\Theta)$, where $P(x|\Theta)$ denotes the likelihood of data x for the model parameter vector Θ, is used for code length $L_{D|M}$ [16]. If the family of models is fixed, such that the description of the family does not depend on the observed data nor the parameters, the cost of encoding a

model may be taken as the number of bits it takes to describe its parameters [16]. The minimized total code length (Eq. (7)) is a measure of goodness of the fitted models. Different types of models can be tried and the results compared.

Let us illustrate the MDL principle with an example. Consider a one-dimensional signal $g(t)$ which was formed by a uniform sampling of a piecewise-constant function $f(t)$ corrupted by additive independent identically distributed (IID) Gaussian noise $N(0,\sigma)$

$$g_k = g(t_k) = f(t_k) + \epsilon \quad k = 1, ..., N \tag{8}$$

$$\epsilon \sim N(0,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp^{n_k^2/(2\sigma^2)} \tag{9}$$

N denotes the number of samples. The amplitude of each sample is quantized to one of the integer values between 0 and A, and Gaussian noise is rounded to the nearest integer q, the precision of the signal values

$$p(n_k) = \int_{[(n_k)]}^{[(n_k)]} \frac{1}{\sqrt{2\pi}\sigma} \exp^{-x^2/(2\sigma^2)} dx \approx \frac{q}{\sqrt{2\pi}\sigma} \exp^{-(f_k - g_k)^2/(2\sigma^2)} \tag{10}$$

when $q < \sigma$. Let q be 1.

Two out of an infinite set of possible descriptions of the signal are:

1. **A pointwise description**. Since any integer l can be encoded with about $\log_2 l$ bits, the total length of encoding equals approximately $N\log_2 A$ bits.
2. **Description in terms of a language**. The language[1] $\mathcal{L}$ involves two components: a deterministic one which specifies the intervals with a constant amplitude, and a stochastic one which encodes the residuals between the model and the data. Let us suppose that the signal is partitioned into n intervals. Each of them is modeled by a constant value $A_1, ..., A_n$. The encoding necessary to describe the model requires

$$L_M = (n-1)\log_2 B + n\log_2 A \approx n\log_2 AB \text{ bits} \tag{11}$$

where the interval boundaries are quantized between 0 and B[2].

The lowest bound on the number of bits that is required to describe data generated by a stochastic process is the negative logarithm with the base 2 of the probability of observing that data. Assuming that the residuals are modeled by an independent identically distributed (IID) Gaussian noise, $N(0,\sigma)$, (σ is encoded to accuracy $\log_2|A|$ bits) truncated to the nearest integer, the obtained optimal code will require

$$L_{D|M} = -\log_2 p(\mathbf{g}-\mathbf{f}) = -\log_2 \prod_{i=1}^{N} p(g_i - f_i) = -\sum_{i=1}^{N} \log_2 p(g_i - f_i) \tag{12}$$

Using Eq. (10) we obtain

$$L_{D|M} = N(\log_2\sigma + \tfrac{1}{2}\log_2 2\pi - \log_2 q) + \tfrac{N-1}{2}\log_2 e \approx N(\log_2\sigma + \tfrac{1}{2}\log_2 2\pi e) \tag{13}$$

The total length to encode both the model and the deviation from the data is $L_M + L_{D|M}$ (Eq. (11) and Eq. (13), respectively).

$$L_M + L_{D|M} \approx n\log_2 AB + N(\log_2\sigma + \tfrac{1}{2}\log_2 2\pi e) \tag{14}$$

Minimizing Eq. (6) answers the question which model (language) is better in the MDL sense:

- a pointwise description: $\mathcal{L}_{m_1} = N\log_2 A$ versus
- description in terms of a language $\mathcal{L}_{M_2} = n\log_2 AB + N(\log_2\sigma + \tfrac{1}{2}\log_2 2\pi e)$.

The solution of this optimization problem supports our intuitive thinking that encoding is efficient if the number of data points described by a model is large and the standard deviation low, while at the same time keeping the complexity of the models small.

3.2. Using models of different scales

Let us now apply this theory to the problem of selecting primitives (models) that operate on different scales. Assume that we have a 1-D signal shown in Fig. 3. For the sake of simplicity we will assume only two types of models since the extension to multiple types of models is straightforward.

1. **Model** $\mathcal{M}_1$ involves two components: parameters V_1 and V_2 specify the constant amplitudes of the corresponding signal. The model can describe only those intervals, which do not contain data points with a deviation exceeding Δ (let $\Delta \to 0$).
2. **Model** $\mathcal{M}_2$ involves two components: parameter V which specifies the constant amplitude of the corresponding signal and a parameter to specify the deviations from the value V. The model can describe only those intervals, which do not contain data points with a deviation exceeding δ (let $\delta \to \infty$).

We will show that the above-defined criteria can decide whether this 1-D signal will be represented:

- either as a piecewise composition of small segments with a minor (or zero) deviation from the model; or
- as a single segment (straight line plus deviations from the model).

1. The description of the signal **in terms of the model** $\mathcal{M}_1$: The signal is partitioned into n intervals. Each of them is modeled by a constant value (V_1 or V_2). The encoding necessary to describe the signal involves only the cost of specifying the amplitude since deviations from the model are smaller than Δ.

$$\mathcal{L}_1 = L_M = (n-1)\log_2 B + n\log_{v_2} V \approx n\log_2 VB \text{ [bits]} \tag{15}$$

where B denotes the resolution of the interval boundaries.

[1] The terms language and model are interchangeable in this case.

[2] For different possible ways of encoding parameters in various applications see [20].

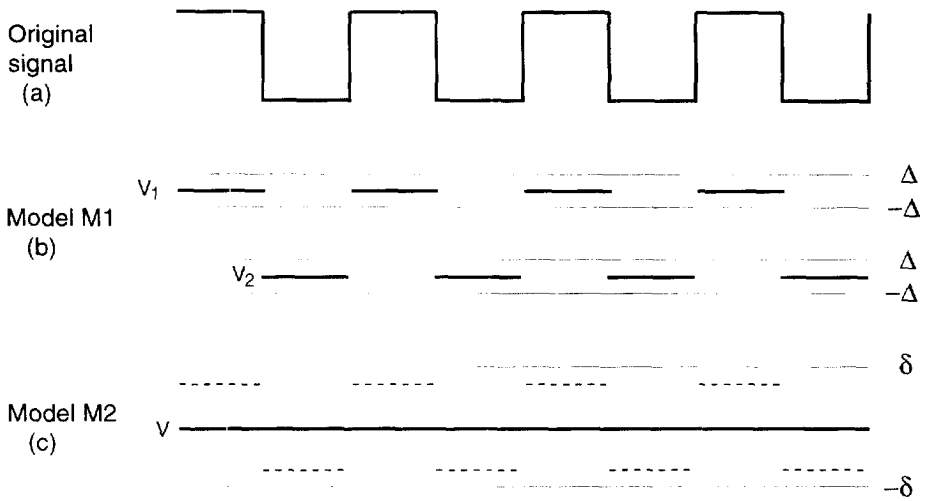

Fig. 3. Description of a signal with two models of different scale. (a) Original 1-D signal. (b) Signal modeled with the model $\mathcal{M}_1$. (c) Signal modeled with the model $\mathcal{M}_2$.

2. The description of the signal **in terms of the model** $\mathcal{M}_2$: In this case we have to specify both the amplitude of the signal V and the deviations from its value. The total length to encode both components is

$$L_2 = L_M + L_{D|M} = \log_2 VB + N\log_2 \delta \ [bits] \qquad (16)$$

Fig. 4 shows the relation between the length of the signal (N), deviation from the constant amplitude (δ), and the corresponding number of bits required to properly encode the signal. The thicker line denotes the number of bits necessary to model the signal in a piecewise manner. The dashed line denotes the number of bits required to model the signal as a single line plus deviations. Several dashed lines are shown for different values of δ.

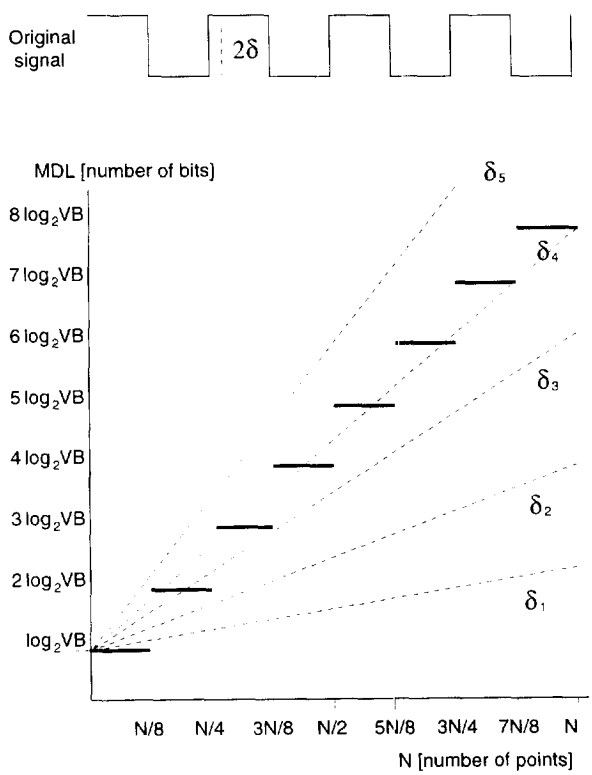

Fig. 4. Relation between MDL, signal length (N), σ, and the type of the models. Thick horizontal lines show the number of bits necessary to model the input signal in a piecewise manner. Dashed lines show the number of bits necessary to model the signal as a line with deviations.

Interestingly enough, the results are in accordance with the intuitive expectations and can be summarized as follows:

- For no deviations or small deviations (e.g. δ_1, δ_2), the signal should be represented as a single straight line, regardless of the number of points (length of the signal).
- When deviations are significant (e.g. δ_5) the signal should always be described as a sum of its piecewise components.
- In the case of deviations indicated by (δ_3, δ_4), we see that the optimal selection of the type of the model (in the MDL sense) depends on the length of the signal. For greater lengths, the signal will be treated as a single model, whereas if we concentrate on a shorter segment of the signal, the optimal description will be given in terms of multiple piecewise models.

If a different encoding system (i.e. language, models) is chosen the result can change accordingly.

4. Examples from real images

To demonstrate the use of the MDL principle we analyze and compare some parametric models recovered from real range images. In the first example we compare surface patches of different granularity. The next example shows superquadric models of different scale. In the third example we compare surface patches and superquadrics which model the same object.

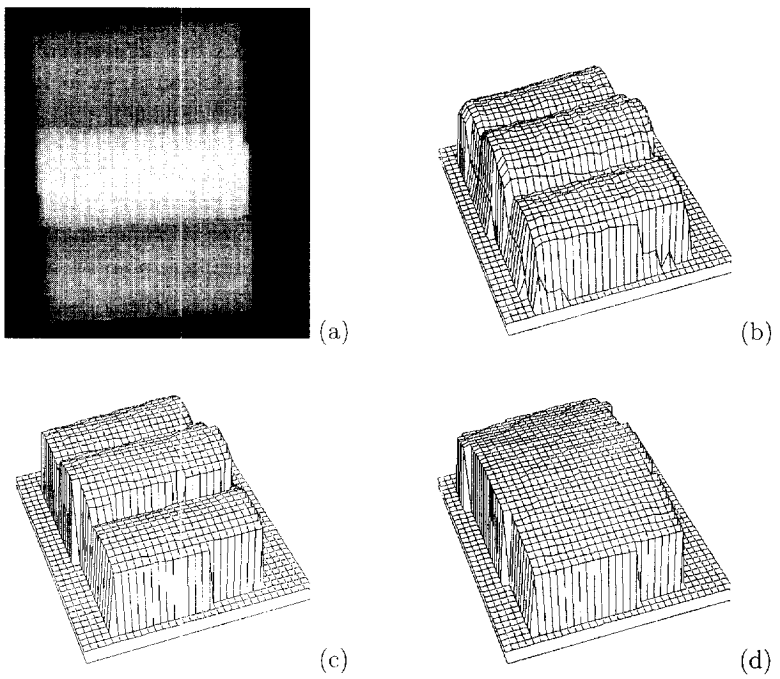

Fig. 5. Three cylindrical surfaces: (a) original range image; (b) 3-D rendition of original data; (c) 3-D rendition of three second-order surfaces; (d) 3-D rendition of a single second-order surface. Increasing the compatibility-constraint smoothes the data to recover one surface.

4.1. Surface patches of different scale

The object shown in Fig. 5 consists of three cylinders joined to form a globally curved structure. We used surfaces to model the range data. When model recovery is performed at the regular scale, we obtain three second-order surfaces (Fig. 5(c)). If we allow higher variance of the noise (e.g. $\sigma \approx 3.2$), i.e. consequently increasing the length of encoding of the deviation, we obtain a single second-order surface describing the 'smoothed' data (Fig. 5(d)).

The models that we have chosen are variable-order bivariate polynomials that are linearly parameterizable in the Euclidean space

$$f(r, \mathbf{a}, \mathbf{x}) = \sum_{0 \le i+j \le r} a_{ij} x^i y^j \tag{17}$$

where the vector $\mathbf{a}$ is defined in the parameter space $\mathcal{A}$. The dimensions of the parameter space depend on the order of the model r which is in our case restricted to $0 \le r \le 2$. Thus, our models consist of planar and biquadric surfaces:

1. *Planar surfaces*: First-order surfaces are written as

$$f(x, y) = a_{10}x + a_{01}y + a_{00} \tag{18}$$

2. *Curved surfaces*: Second-order surfaces are written as

$$f(x, y) = a_{20}x^2 + 2a_{11}xy + a_{02}y^2 + 2a_{10}x + 2a_{01}y + a_{00} \tag{19}$$

A recovered surface patch model consists of the domain of the model (a set of N domain points (x_i, y_i) that belong to the model) and the set of six model parameters $\mathbf{a}$ (see also the section on model selection).

After Eq. (15) the length of the encoding for the three second-order surfaces that fit well to the data (points deviate less than Δ, where $\Delta \to 0$) can be computed as follows:

$$\mathcal{L}_1 = L_M = N \cdot 2\log_2 D + 3 \cdot 6\log_2 A \text{ [bits]} \tag{20}$$

where N is the total number of range points in the image which are represented by the three models, $\log_2 D$ denotes the average number of bits needed to specify a coordinate value D and $\log_2 A$ denotes the average number of bits to specify a parameter value A of the model.

The length of the coarser description requires the specification of a single second-order surface as well as the deviations of N individual range points from its surface (Eq. (16)):

$$\mathcal{L}_2 = L_M + L_{D|M} = \tag{21}$$

$$= N \cdot 2\log_2 D + 6\log_2 A + N\log_2 \delta \text{ [bits]} \tag{22}$$

where $\log_2 \delta$ denotes the average number of bits needed to specify the deviation. The decision as to which representation to use in this example now rests on how important the information on deviations from the model is. For a coarse description often only the information that all deviations do not exceed a certain value will suffice.

4.2. Superquadrics of different scale

The object shown in Fig. 6 is assembled out of a sphere and a cylinder. When model recovery is done at a regular scale two volumetric models are obtained. Again, if the compatibility constraint is loosened, a single tapered cylinder, which is a rough description of the object, is obtained.

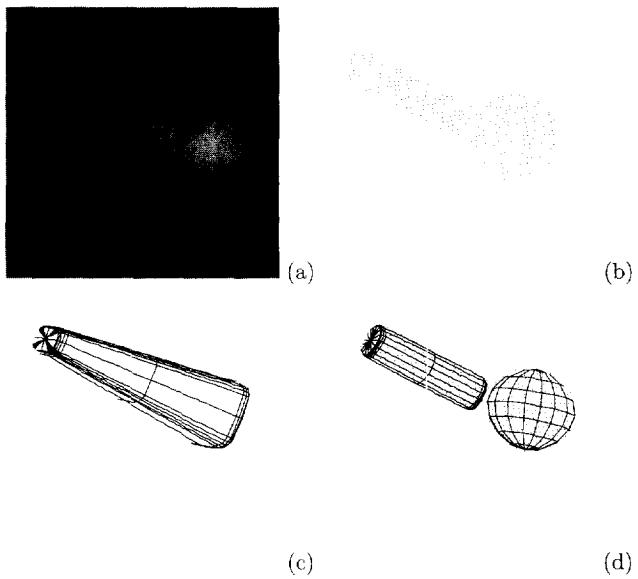

Fig. 6. A sphere attached to a cylinder: (a) intensity image; (b) range image; (c) a tapered superquadric; (d) two superquadrics (a sphere and a cylinder) superimposed on the original range data. Depending on the selected level of description, the system can recover descriptions of different granularity.

The models in this case are superquadrics which have been used extensively for shape representation in computer graphics and vision as part-level models [21–23]. The superquadric model in an object centered coordinate system is defined by the following equation:

$$F(x,y,z)=\left(\left(\frac{x}{a_1}\right)^{\frac{2}{\epsilon_2}}+\left(\frac{y}{a_2}\right)^{\frac{2}{\epsilon_2}}\right)^{\frac{\epsilon_2}{\epsilon_1}}+\left(\frac{z}{a_3}\right)^{\frac{2}{\epsilon_1}} \quad (23)$$

Superquadrics are an extension of basic quadric surfaces and solids. Exponents ϵ_1 and ϵ_2 control the squareness of edges so that ellipsoid, cylinder, parallelepiped and shapes in-between can be represented. Modeling capabilities of superquadrics can be enhanced by deforming them with parametric global deformations, such as linear tapering which requires just one additional parameter. In general position, an additional six parameters are needed, three for position (p_x, p_y, p_z) and three for orientation (ϕ, τ, ψ) of the superquadric. For details of the superquadric recovery method used to obtain the models in Fig. 6, see [23,6,24].

After Eq. (15) the length of the encoding for the two superquadrics that fit well to the data (points deviate less than Δ) can be computed as follows:

$$\mathcal{L}_1 = L_M = 2{\cdot}11 \log_2 A \text{ [bits]} \quad (24)$$

where A are the values of the superquadric parameters. The length of the coarser description requires the specification of the single tapered superquadric as well as the deviations of N individual range points from its surface (Eq. (16)):

$$\mathcal{L}_2 = L_M + L_{D|M} = (11+1)\log_2 A + N\log_2 \delta \text{ [bits]} \quad (25)$$

where the additional twelveth parameter comes from tapering. There is no need to specify the boundaries of superquadric models since they are implicit, compact and self-contained.

Again, the particular decision as to which representation to use in this example now rests on how large the deviations from the model are that we can tolerate.

4.3. Different types of models

In this example we compare the surface-level and volumetric-level description of an L-shaped object (Fig. 7). The models are the same as in the first and second example, respectively.

After Eq. (15) the length of the encoding for the surface-level representation which consists of five planar patches each requiring three parameters is:

$$\mathcal{L}_1 = L_M = N{\cdot}2\log_2 D + 5{\cdot}3\log_2 A \text{ [bits]} \quad (26)$$

and for the volumetric-level description:

$$\mathcal{L}_1 = L_M = 2{\cdot}11 \log_2 A \text{ [bits]} \quad (27)$$

Given that the data fits well to the models (points deviate less than Δ and $\Delta \rightarrow 0$) in both cases, the volumetric description is, in this particular example, shorter than the surface description.

An interpretation with surface and volumetric models of a more complex scene is shown in Figs 8–11. To avoid the question of when to perform the selection, we postponed this decision until all the surface and volumetric models were fully grown. In short, we first independently recovered the surfaces using the recover-and-select paradigm, then did the same for the volumetric models, and finally selected models from both sets of models. Due to a more compact

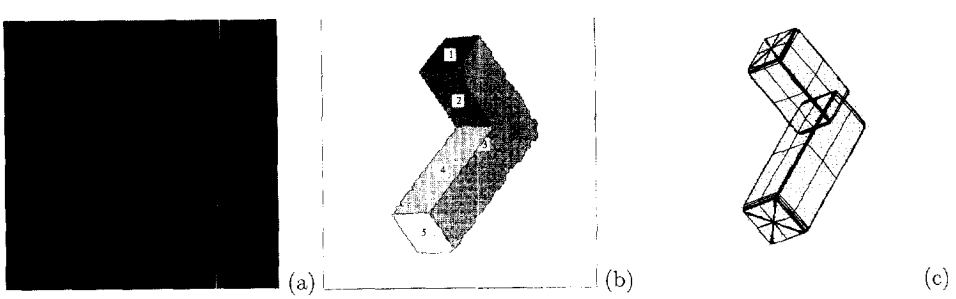

Fig. 7. L-shaped object: (a) range image; (b) surface-level description; (c) volumetric-level description.

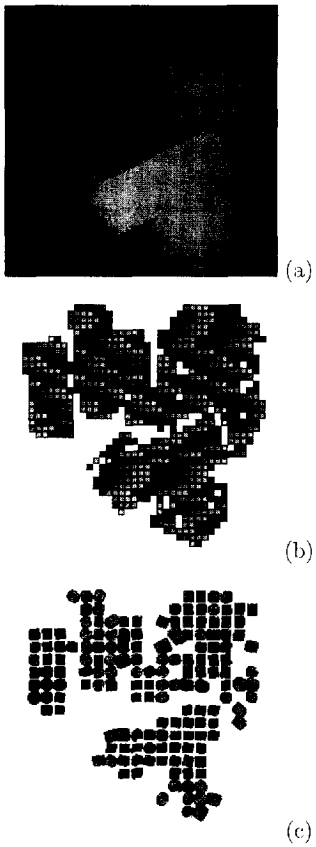

Fig. 8. (a) Range image, (b) initial surface models and (c) initial volumetric models.

representation, superquadrics are generally preferred over surface models. However, only those superquadrics with a reasonably good fit were selected for the final representation. Where the fit is poor or the range data is not covered with superquadrics, surface models are selected.

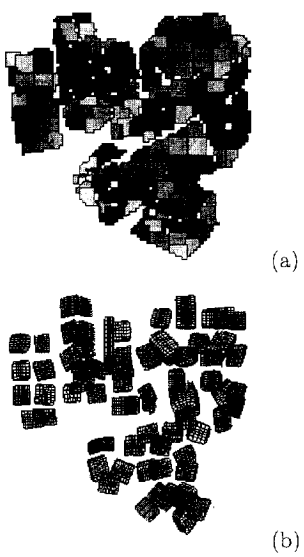

Fig. 9. First recover-and-select iterations for (a) surface models and (b) volumetric models.

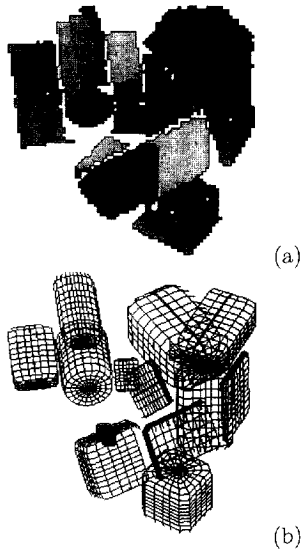

Fig. 10. Second recover-and-select iterations for (a) surface models and and (b) volumetric models.

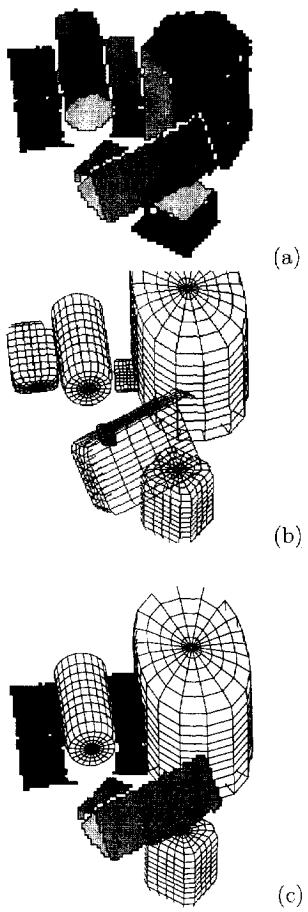

Fig. 11. Independently recovered (a) surface models and (b) volumetric models. Final result (c).

5. Conclusion

In this paper we propose an efficient architecture for selective image modeling. Models of different scale and type are reconstructed in parallel. We show that the redundant representation can be efficiently pruned using the criterion of Minimal Description Length. In this way the optimal scale for perceiving a scene is selected since due to the *bounded rationality principle* only the most important information should be passed on to higher levels of the system.

We have implemented the general concept in an object oriented framework. This enables the experimentation with a variety of different models. The basic research issues that remain are related to the questions of computational complexity in the presence of multiple models. The question is whether we can invoke the selection procedure even before the models are fully recovered and thus eliminate some of the superfluous models. While this strategy turned out to be very effective in the case of models of one type [5], in the case of multiple models this poses certain problems. It is easy to show that the selection process over volumetric and surface models before these descriptions are fully grown can produce spurious results [25]. This can happen because during the growth, a single volumetric model might locally model the image much better than a set of corresponding surface patches and consequently surface patches are rejected. However, if both the volumetric models and the surface patches were fully grown, the surface patches would be selected. This is even more so if a simple greedy algorithm is used for the selection.

Acknowledgements

This work was supported by the Ministry of Science and Technology of the Republic of Slovenia (Projects J2-6187 and J2-8829), European Union Copernicus Program (Grant 1068 RECCAD), and by U.S.-Slovene Joint Board (Project #95-158).

References

[1] R. Bajcsy, Active perception, Proceedings of the IEEE 76 (8) (1988) 996–1005.

[2] Y. Aloimonos, Purposive and qualitative active vision, Proceedings of the 10th International Conference on Pattern Recognition, Atlantic City, NJ, June 1990, pp. 346–360.

[3] L. Stark, Recognizing object function through reasoning about 3-D shape and dynamic physical properties, Proceedings IEEE Computer Vision and Pattern Recognition Conference, Seattle, WA, 1994, pp. 546–553.

[4] D. Marr, Vision, Freeman, San Francisco, 1982.

[5] A. Leonardis, A. Gupta, R. Bajcsy, Segmentation of range images as the search for geometric parametric models, International Journal of Computer Vision 14 (1995) 253–277.

[6] A. Leonardis, F. Solina, A. Macerl, A direct recovery of volumetric models in range images using recover-and-select paradigm, in Jan-Olof Eklundh (Ed.), Computer Vision-ECCV'94, Volume I, Stockholm, Sweden, Springer, Berlin, 1994, pp. 309–318.

[7] A.P. Pentland, Part segmentation for object recognition, Neural Computation 1 (1989) 82–91.

[8] A. Leonardis, Image analysis using parametric models: model-recovery and model-selection paradigm, PhD Dissertation, University of Ljubljana, Faculty of Electrical Engineering and Computer Science, Ljubljana, Slovenia, 1993.

[9] Y.G. Leclerc, Constructing simple stable descriptions for image partitioning, International Journal of Computer Vision 3 (1989) 73–102.

[10] S.W. Zucker, Region growing: childhood and adolescence, Computer Graphics Image Processing 5 (1976) 382–399.

[11] A.F. Bobick, R.C. Bolles, The representation space paradigm of concurrent evolving object descriptions, IEEE Transactions on Pattern Recognition and Machine Intelligence 14 (2) (1992) 146–156.

[12] H. Simon, Theories of bounded rationality, in C. McGuire, R. Radner (Eds.), Decision and Organization, Chapter 8, pp. 161–176, North-Holland, Amsterdam, 1972.

[13] J. Hochberg, Perceptual Organization, chapter Levels of Perceptual Organization. Lawrence Erlbaum Associates, New Jersey, 1981, pp. 255–276.

[14] C. Shannon, A mathematical theory of communication, Bell Systems Technical Journal 27 (1948) 379–423.

[15] J. Rissanen, Modeling by shortest data description, Automatica 14 (1978) 465–471.

[16] J. Rissanen, A universal prior for integers and estimation by minimum description length, The Annals of Statistics 11 (1983) 412–431.

[17] J. Rissanen, Minimum-Description-Length Principle. In Encyclopedia of Statistical Sciences, Vol. 5, pp. 523–527, John Wiley and Sons, New York, 1987.

[18] P. Fua, A.J. Hanson, Objective functions for feature discrimination, Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, pp. 1596–1602, Morgan Kaufman, Detroit, MI, August 1989.

[19] K.C. Keeler, Map representations and optimal encoding for image segmentation, Technical Report CICS-TH-292, Center for Intelligent Control Systems, March 1991.

[20] M. Li, P. Vitanyi, An introduction to Kolmogorov complexity and its applications, Second edn, Springer, New York, 1997.

[21] A.P. Pentland, Perceptual organization and the representation of natural form, Artificial Intelligence 28 (2) (1986) 293–331.

[22] F.P. Ferrie, J. Lagarde, P. Whaite, Darboux frames, snakes and superquadrics: geometry from the bottom up, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (8) (1993) 771–784.

[23] F. Solina, R. Bajcsy, Recovery of parametric models from range images: the case for superquadrics with global deformations, IEEE Trans. on Pattern Analysis and Machine Intelligence 12 (2) (1990) 131–147.

[24] F. Solina, A. Leonardis, Shape decomposition using part-models of different granularity, in C. Arcelli, L.P. Cordella, G. Sanniti di Baja (Eds.), Aspects of Visual Form Processing, World Scientific, Singapore, 1994.

[25] A. Leonardis, A. Jaklič, B. Kverh, F. Solina, Simultaneous recovery of surface and superquadric models, in Axel Pinz (Ed.), Pattern Recognition 1996, Proceedings of 20th Workshop of the Austrian Pattern Recognition Group (OAGM/AAPR), Shriftenreihe der OCG, Band 90, pp. 27–36, Wien, Munchen, 1996. Osterreichische Computer Gesellschaft, R. Oldenburg.