２０００年情報論的学習理論ワークショップ
2000 Workshop on Information-Based Induction Sciences (IBIS2000)
Izu, Japan, July 17-18, 2000.

Geometric MDL and Its Media Applications

Kenichi Kanatani* Chikara Matsunaga†

Abstract: Based on the duality that the asymptotic properties of statistical inference in the limit of an infinite number of observations hold for geometric fitting in the limit of infinitesimal perturbations, we define the *geometric MDL* as the dual to *Rissanen's MDL* and give an information-theoretic interpretation. Doing numerical simulations and real image virtual studio experiments, we observe that the geometric AIC and the geometric MDL have different characteristics.

1. Introduction

Model selection has been one of the central themes in the traditional statistical domains such as agriculture, medicine, politics, and economics. Recently, interest in model selection for computer vision and robotics applications has been quickly growing, partly due to the introduction of the *geometric AIC* [4, 5], which is a variant of Akaike's *AIC* (*Akaike information criterion*) [1]. Another well known information criterion in statistics is Rissanen's *MDL* (*minimum description length*) [13, 14], which is known to be superior to Akaike's AIC in some respects. Naturally, there have been attempts to apply MDL to computer vision and robotics problems [3, 9, 12, 16].

The purpose of this paper is to point out that some of these attempts are based on a misunderstanding of the fundamental structure of geometric problems. The key fact is the duality that *the asymptotic properties of statistical inference in the limit of an infinite number of observations hold for geometric fitting in the limit of infinitesimal perturbations*. In this paper, we define the *geometric MDL* as the dual to Rissanen's MDL and give an information-theoretic interpretation. We then do numerical simulations and real image virtual studio experiments and observe that the geometric AIC and the geometric MDL have different characteristics.

2. Geometric MDL

2.1 Geometric fitting

Geometric fitting is defined as follows [4, 6]. Let N m-dimensional vectors $\{\boldsymbol{x}_\alpha\}$ be given as data. Each $\boldsymbol{x}_\alpha$ is regarded as perturbed from its true value $\bar{\boldsymbol{x}}_\alpha$ by $\Delta\boldsymbol{x}_\alpha$, which we assume is an independent Gaussian random variable of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{x}_\alpha]$. The true value $\bar{\boldsymbol{x}}_\alpha$ is assumed to satisfy r constraint equations

$$F^{(k)}(\bar{\boldsymbol{x}}_\alpha, \boldsymbol{u}) = 0, \qquad k = 1, ..., r, \tag{1}$$

parameterized by a p-dimensional vector $\boldsymbol{u}$. We assume that the covariance matrix $V[\boldsymbol{x}_\alpha]$ is known only up to scale and write

$$V[\boldsymbol{x}_\alpha] = \epsilon^2 V_0[\boldsymbol{x}_\alpha]. \tag{2}$$

The ML-estimator of the parameter $\boldsymbol{u}$ is obtained by minimizing the sum of the square Mahalanobis distances from the data points to their true positions associated with the normalized covariance matrices $V_0[\boldsymbol{x}_\alpha]$

$$J = \sum_{\alpha=1}^{N} (\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}, V_0[\boldsymbol{x}_\alpha]^{-1}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}})) \tag{3}$$

subject to the constraint (1). Throughout this paper, the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is denoted by $(\boldsymbol{a}, \boldsymbol{b})$.

2.2 Rissanen's MDL and its dual

Rissanen's *MDL* (*minimum description length*) [13, 14] measures the goodness of a model by the code length of its description. Since it is based on asymptotic analysis of probability densities in the limit of infinite data length, it cannot to be applied to geometric models, to which no stochastic meanings or asymptotic analysis can be given. This is a big contrast to Akaike's AIC in that it can directly be applied to geometric models because it is based on expectation evaluation [6].

The asymptotic nature of Rissanen's MDL is implicitly shown in the term $k \log n$, where k is the number of parameters and n is the "data length". What does the "data length" exactly mean? Suppose we are given ten points (x_1, y_1), ..., (x_{10}, y_{10}) in two dimensions. Is the data length $n = 10$ as the number of the points? Or is it $n = 20$ as the number of the real numbers? Or is it $n = 1$ as a point in the 20-dimensional space $\mathcal{R}^{20}$?

If we closely examine Rissanen's argument, it turns out that the data length n is a number such that $1/n$ the Fisher information matrix converges to a constant matrix as $n \to \infty$. In the above example, we can view n as 10 or 5, because quantities that have smaller orders than $O(\log n)$ in the limit $n \to \infty$ are ignored: $\log 2n = \log n + \log 2 \approx \log n$. In other words, the data

* Department of Computer Science, Gunma University, Kiryu, Gunma, 376-8515 Japan, Tel: (0277)30-1841, Fax: (0277)30-1801, E-mail: kanatani@cs.gunma-u.ac.jp.

† Broadcast Division, FOR-A Co. Ltd., 2-3-3 Ohsaku, Sakura, Chiba 285-0802 Japan, Tel: (043)498-1239, Fax: (043)498-2223, E-mail: matsunaga@for-a.co.jp.

length, which critically characterize Rissanen's MDL, is an asymptotic concept. It follows that Rissanen's MDL is to be applied to such a large n that $\log cn = \log n + \log c \approx \log n$ for a constant c.

This is an inevitable consequence of the fact that encoding a real number requires an infinite code length. Rissanen's idea is to quantize the parameters to obtain a finite code length, taking into account the fact that real numbers cannot be estimated completely [14]. The quantization width is determined by attainable estimation accuracy, which in turn is determined by the data length n. Since the code length diverges as $n \to \infty$, asymptotic approximation comes into play. In this sense, the "minimum description length" actually means the minimum "growth rate" of the description length.

As pointed out in [6], the number of observations is always $n = 1$ in geometric fitting. Hence, Rissanen's MDL, which makes sense only in the limit $n \to \infty$, does not make sense for geometric fitting. On the other hand, if we recall the duality between the limit $n \to \infty$ of an infinite number of observations and the limit $\epsilon \to 0$ of infinitesimal perturbation, we can define the *dual* to Rissanen's MDL by considering the "growth rate" of the description length as $\epsilon \to 0$.

Note that the noise level ϵ in geometric fitting effectively decreases to $O(1/\sqrt{n})$ as the number of hypothetical observations increases [6], which implies $n = O(1/\epsilon^2)$. Rissanen's MDL consists of the negative logarithmic likelihood and $(k/2)\log n$. The negative logarithmic likelihood for Gaussian noise is $J/2\epsilon^2$. Hence, we can define the dual to Rissanen's MDL in the form

$$\text{G-MDL} = \hat{J} - (Nd + p)\epsilon^2 \log \epsilon^2, \tag{4}$$

which we call the *geometric MDL*. Since the noise level ϵ is not a model parameter [6], it should be estimated from the general model as follows [4]:

$$\hat{\epsilon}^2 = \frac{\hat{J}}{rN - p} \tag{5}$$

2.3 Information-theoretic interpretation

We now show that eq. (4) can indeed be interpreted to be the minimum description length (or its growth rate) of the geometric fitting problem. The problem is specified by (i) the data $\{\boldsymbol{x}_\alpha\}$, (ii) their true positions $\{\bar{\boldsymbol{x}}_\alpha\}$, and (iii) the parameter $\boldsymbol{u}$ of the model, all of which should be encoded.

(i) We encode the data $\{\boldsymbol{x}_\alpha\}$. The probability density of the data $\{\boldsymbol{x}_\alpha\}$ has the form

$$P = \frac{e^{-J/2\epsilon^2}}{(2\pi\epsilon^{2m})^N \prod_{\alpha=1}^N \det V_0[\boldsymbol{x}_\alpha]}. \tag{6}$$

Encoding the data $\{\boldsymbol{x}_\alpha\}$ in a prefix code for this probability density requires at least the code length

$$-\log P = \frac{J}{2\epsilon^2} + N\log(2\pi\epsilon^{2m}) + \sum_{\alpha=1}^N \log\det V_0[\boldsymbol{x}_\alpha]. \tag{7}$$

Information theoretically, the base of the logarithm should be 2, but it can be viewed as the natural logarithm if the code length is measured in $\log_2 e$ bits. Note that the last two terms on the left-hand side of eq. (7) are constants that do not depend on the individual (geometric) models.

(ii) We encode the true positions $\{\bar{\boldsymbol{x}}_\alpha\}$. Since they cannot be determined exactly as long as noise exists, we encode their ML-estimators $\{\hat{\boldsymbol{x}}_\alpha\}$ [14]. They are the projections of the data $\{\bar{\boldsymbol{x}}_\alpha\}$ onto the d-dimensional model manifold $\mathcal{S}$ with the minimum sum of squared Mahalanobis distances (*Mahalanobis projections* [4]). Their covariance matrices have the following form [4]:

$$V[\hat{\boldsymbol{x}}_\alpha] = \epsilon^2 V_0[\hat{\boldsymbol{x}}_\alpha] + O(\epsilon^4). \tag{8}$$

Here, $V_0[\hat{\boldsymbol{x}}_\alpha]$ is an $m \times m$ singular positive semi-definite symmetric matrix with rank d that has the form

$$V_0[\hat{\boldsymbol{x}}_\alpha] = V_0[\boldsymbol{x}_\alpha] - \sum_{k,l=1}^r W_\alpha^{(kl)} (V[\boldsymbol{x}_\alpha]\nabla_{\mathbf{x}} F_\alpha^{(k)})(V[\boldsymbol{x}_\alpha]\nabla_{\mathbf{x}} F_\alpha^{(k)})^\top, \tag{9}$$

where $W_\alpha^{(kl)}$ is the (kl) element of the inverse of the $r \times r$ matrix whose (kl) element equals $(\nabla_{\mathbf{x}} F^{(k)}, V[\boldsymbol{x}_\alpha]\nabla_{\mathbf{x}} F^{(l)})$.

The error $\Delta\hat{\boldsymbol{x}}_\alpha$ in $\hat{\boldsymbol{x}}_\alpha$ has the order $\sqrt{E[\|\Delta\hat{\boldsymbol{x}}_\alpha\|^2]} = \sqrt{\mathrm{tr} V[\hat{\boldsymbol{x}}_\alpha]} = O(\epsilon)$, where tr denotes the trace. It follows that it is sufficient to encode $\hat{\boldsymbol{x}}_\alpha$ up to accuracy $O(\epsilon)$. So, we define a grid of mesh size $c\epsilon$ over the manifold $\mathcal{S}$ with some constant c. Since $\mathcal{S}$ is a d-dimensional manifold, the necessary code length for specifying a grid point is constant $-\log(c\epsilon)^d$ = constant $- d\log\epsilon$. Hence, the total code length for the N data points is constant $- Nd\log\epsilon$.

(iii) We encode the parameter $\boldsymbol{u}$. Since this, too, cannot be determined exactly in the presence of noise, we encode its ML-estimator $\hat{\boldsymbol{u}}$. Its covariance matrix has the following form [4]:

$$V[\hat{\boldsymbol{u}}] = \epsilon^2 V_0[\hat{\boldsymbol{u}}] + O(\epsilon^4). \tag{10}$$

Here, $V_0[\hat{\boldsymbol{u}}_\alpha]$ is an $p \times p$ positive definite symmetric matrix that has the form

$$V_0[\hat{\boldsymbol{u}}] = \Big(\sum_{\alpha=1}^N \sum_{k,l=1}^r W_\alpha^{(kl)} (\nabla_{\mathbf{u}} F_\alpha^{(k)})(\nabla_{\mathbf{u}} F_\alpha^{(l)})^\top\Big)^{-1}. \tag{11}$$

As in the case of $\hat{\boldsymbol{x}}_\alpha$, it is sufficient to encode $\hat{\boldsymbol{u}}$ up to accuracy $O(\epsilon)$. So, we define a grid of mesh size $C\epsilon$ in the domain $\mathcal{R}^p$ of $\boldsymbol{u}_\alpha$ with some constant C. The necessary code length for specifying a grid point is constant $-\log(c\epsilon)^p$ = constant $- p\log\epsilon$.

Ignoring constant terms that do not depend on the individual models and noting that replacing J by $\hat{J}$ introduces only errors of higher order in ϵ, we conclude that the total code length is

$$\frac{\hat{J}}{2\epsilon^2} - (Nd + p)\log\epsilon. \tag{12}$$

Multiplying this by $2\epsilon^2$, we obtain eq. (4). Recall that ϵ is *not* a model parameter. Multiplication by a positive expression in ϵ does not affect relative comparison of models, because ϵ is common to all the models [6].

2.4 Optimal quantization

The above analysis is a rough approximation. Following Rissanen [14], we now give a more precise evaluation. Eqs. (8) and (10) imply that eq. (3) can be written as follows [4]:

$$J = J_0 + \sum_{\alpha=1}^{N} (\boldsymbol{x}_\alpha - \hat{\boldsymbol{x}}, V_0[\hat{\boldsymbol{x}}_\alpha]^-(\boldsymbol{x}_\alpha - \hat{\boldsymbol{x}})) + (\boldsymbol{u} - \hat{\boldsymbol{u}}, V_0[\hat{\boldsymbol{u}}]^{-1}(\boldsymbol{u} - \hat{\boldsymbol{u}})) + O(\epsilon^3). \quad (13)$$

Here, J_0 is the value of J evaluated at $\boldsymbol{x}_\alpha = \hat{\boldsymbol{x}}$ and $\boldsymbol{u} = \hat{\boldsymbol{u}}$, and $V_0[\hat{\boldsymbol{x}}_\alpha]^-$ is the (Moore-Penrose) generalized inverse of $V_0[\hat{\boldsymbol{x}}_\alpha]^-$.

We encoded $\hat{\boldsymbol{u}}$ by defining a grid of mesh size $C\epsilon$ in the domain $\mathcal{R}^p$ of $\boldsymbol{u}$. We now let the mesh size vary from cell to cell in such a way that the resulting code length is minimized. If we quantize the ith coordinate axis of $\mathcal{R}^p$ into intervals of width δu_i, $i = 1, ..., p$, the code length for specifying a vertex of the resulting grid is constant $-\sum_{i=1}^{p} \log \delta u_i$.

From eq. (13), we see that the terms that involve $\boldsymbol{u}$ in the code length given by eq. (7) are

$$\frac{(\delta\boldsymbol{u}, V_0[\hat{\boldsymbol{u}}]^{-1}\delta\boldsymbol{u})}{2\epsilon^2} - \sum_{i=1}^{p} \log \delta u_i, \quad (14)$$

where we define the vector $\delta\boldsymbol{u} = (\delta u_i)$. We then determine δu_i that minimizes eq. (14). Differentiating eq. (14) with respect to δu_i and letting the result be zero, we obtain

$$\frac{1}{\epsilon^2}\left(V_0[\hat{\boldsymbol{u}}]^{-1}\delta\boldsymbol{u}\right)_i = \frac{1}{\delta u_i}. \quad (15)$$

If we take the coordinate system of $\mathcal{R}^p$ in such a way that $V_0[\hat{\boldsymbol{u}}]$ is diagonalized, we obtain the solution $\delta u_i = \epsilon\sqrt{\lambda_i}$, where λ_i is the ith eigenvalue of $V_0[\hat{\boldsymbol{u}}]$. Substituting this into eq. (14) and noting that $\prod_{i=1}^{p} \lambda_i = \det V_0[\hat{\boldsymbol{u}}]$, we have

$$\frac{p}{2} - \frac{p}{2}\log\epsilon^2 - \frac{1}{2}\log\det V_0[\hat{\boldsymbol{u}}]. \quad (16)$$

Since $-\log\epsilon^2 \gg 1$ when ϵ is small, we pick out only this dominant term, obtaining the term $-p\epsilon^2\log\epsilon^2$ in eq. (4) after division by $2\epsilon^2$.

Similarly, defining a grid of variable mesh size over the manifold $\mathcal{S}$ and optimizing it, we obtain the term $-Nd\epsilon^2\log\epsilon^2$ in eq. (4) divided by $2\epsilon^2$. Multiplying the resulting total expression by $2\epsilon^2$, we arrive at eq. (4). From our analysis, we observe that the geometric MDL is to be applied to such a small ϵ that $-\log c\epsilon = -\log\epsilon - \log c \approx -\log\epsilon$ for a constant c.

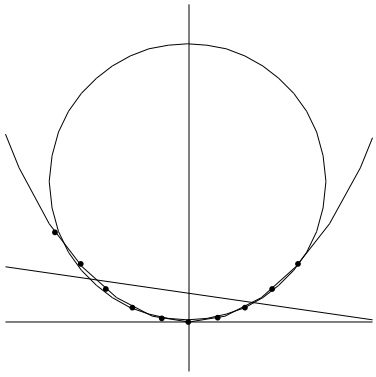


Figure 1: Fitting a line, a circle, and an ellipse.

3. Numerical Simulation Experiments

3.1 Detection of circles and lines

We consider an ellipse that is tangent to the x axis at the origin O with radius 50 in the y direction and eccentricity $1/\beta$. On it, we take eleven points that have equally spaced x coordinates. Adding random Gaussian noise of standard deviation σ to the x and y coordinates of each point independently, we fit an ellipse, a circle, and a line in a statistically optimal manner using a technique called *renormalization* [4, 7, 8]. Fig. 1 shows one instance for $\beta = 2.5$ and $\sigma = 0.1$. Note that a line and a circle are both special cases (degeneracies) of an ellipse.

Lines, circles, and ellipses all define one-dimensional (geometric) models with 2, 3, and 5 degrees of freedom, respectively. Hence, their geometric AIC and the geometric MDL for N points are given as follows:

$$\begin{aligned}
\text{G-AIC}_l &= \hat{J}_l + 2(N+2)\sigma^2, \\
\text{G-AIC}_c &= \hat{J}_c + 2(N+3)\sigma^2, \\
\text{G-AIC}_e &= \hat{J}_e + 2(N+5)\sigma^2, \\
\text{G-MDL}_l &= \hat{J}_l - (N+2)\sigma^2\log\sigma^2, \\
\text{G-MDL}_c &= \hat{J}_c - (N+3)\sigma^2\log\sigma^2, \\
\text{G-MDL}_e &= \hat{J}_e - (N+5)\sigma^2\log\sigma^2. \quad (17)
\end{aligned}$$

The subscripts l, c, and e refer to lines, circles, and ellipses, respectively. For each β, we compare the geometric AIC and the geometric MDL of the fitted line, circle, and ellipse and choose the one that has the smallest geometric AIC or the smallest geometric MDL.

Fig. 2(a) shows the percentage of choosing a line for $\sigma = 0.01$ after 1000 independent trials for each β in the neighborhood of $\beta = 0$. If there were no noise, it would be 0% for $\beta \neq 0$ and 100% for $\beta = 0$. In the presence of noise, the geometric AIC gives a sharp peak, indicating a high capability of distinguishing a line from an ellipse. However, it judges a line to be an ellipse with some probability. In contrast, the geometric MDL judges a line to be a line at almost 100% for small noise but judges an ellipse to be a line over a wide range of β.

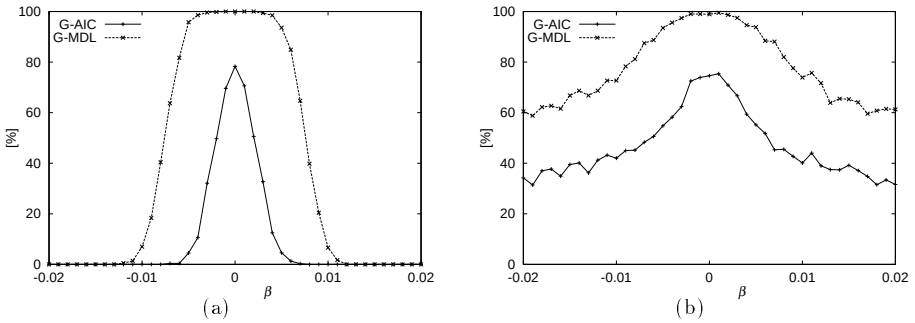


Figure 2: The rate (%) of detecting a line by the geometric AIC (solid lines) and the geometric MDL (dashed lines) with (a) the true noise level and (b) the estimated noise level.

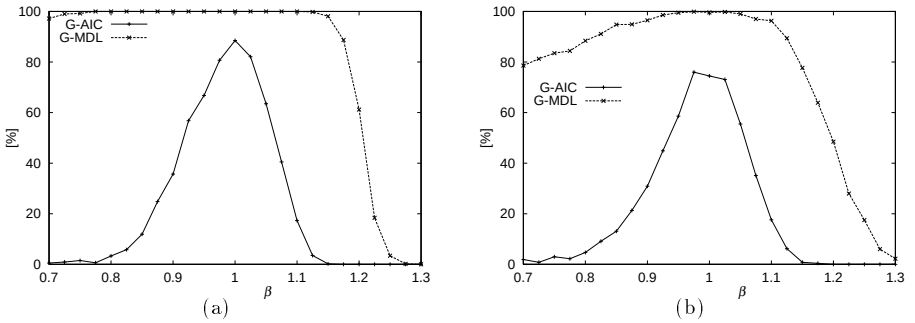


Figure 3: The rate (%) of detecting a circle by the geometric AIC (solid lines) and the geometric MDL (dashed lines) with (a) the true noise level and (b) the estimated noise level.

In Fig. 2(a), we used the true value of the noise variance σ^2 in computing the geometric AIC and the geometric MDL. If it is unknown, it can be estimated from the residual of the general ellipse model. Fig. 2(b) shows the result using the estimated σ^2. Although the sharpness is somewhat lost, we observe similar performance characteristics of the geometric AIC and the geometric MDL.

Fig. 3 shows the percentage of choosing a circle for $\sigma = 0.01$ in the neighborhood of $\beta = 1$. If there were no noise, it would be 0% for $\beta \neq 1$ and 100% for $\beta = 1$. In the presence of noise, as we see, it is difficult to distinguish a small circular arc from a small elliptic arc for $\beta < 1$. Yet, the geometric AIC can detect a circle very sharply, although it judges a circle to be an ellipse with some probability. In contrast, the geometric MDL almost always judges an ellipse to be a circle for $\beta < 1.1$.

3.2 Detection of space lines

Unlike the stochastic model for statistical inference, the geometric model is a manifold defined by the constraint in the data space. Hence, its *dimension* plays an important role as well as its degree of freedom [4, 5]. Lines and curves in two dimensions all have dimension 1. Here, we compare two models that have different dimensions: we fit a line and a plane to points in three dimensions.

We randomly take eleven points in a rectangular region $[0, 10] \times [-1, 1]$ in the xy plane and enlarge them A times in the y direction. Adding random Gaussian noise of standard deviation σ to the x, y, and z coordinates of each point independently, we fit a line and a plane in a statistically optimal manner (see [4] for the details of the computation). Note that a space line is a degeneracy of a plane.

A space line is a one-dimensional model with four degrees of freedom; a plane is a two-dimensional model with three degrees of freedom. Hence, their geometric AIC and geometric MDL have the following form:

$$\begin{aligned} \text{G-AIC}_l &= \hat{J}_l + 2(N+4)\sigma^2, \\ \text{G-AIC}_p &= \hat{J}_p + 2(2N+3)\sigma^2, \\ \text{G-MDL}_l &= \hat{J}_l - (N+4)\sigma^2 \log \sigma^2, \\ \text{G-MDL}_p &= \hat{J}_p - (2N+3)\sigma^2 \log \sigma^2. \end{aligned} \tag{18}$$

The subscripts l and p refer to lines and planes, respectively. For each A, we compare the geometric AIC and the geometric MDL of the fitted line and plane and choose the one that has the smaller geometric AIC or the smallest geometric MDL.

Fig. 4(a) shows the percentage of choosing a line for $\sigma = 0.01$ after 1000 independent trials for each A in the neighborhood of $A = 0$. If there were no noise, it would be 0% for $A \neq 0$ and 100% for $A = 0$. In the presence of noise, the geometric AIC has a high capability of distinguishing a line from a plane but judges a line

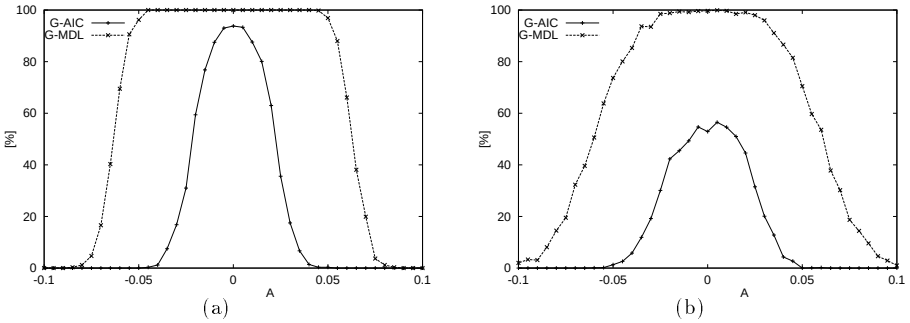


Figure 4: The rate (%) of detecting a space line by the geometric AIC (solid lines) and the geometric MDL (dashed lines) with (a) the true noise level and (b) the estimated noise level.

to be a plane with some probability. In contrast, the geometric MDL judges a line to be a line at almost 100% for small noise but judges a plane to be a line over a wide range of A.

In Fig. 4(a), we used the true value of the noise variance σ^2. Fig. 4(b) shows the corresponding result using its estimate obtained from the residual of the general plane model. We observe somewhat degraded but similar performance characteristics.

3.3 The geometric AIC vs. the geometric MDL

In sum, the geometric AIC has a higher capability for distinguishing degeneracy than the geometric MDL, but the general model is chosen with some probability when the true model is degenerate. In contrast, the percentage for the geometric MDL to detect degeneracy when the true model is degenerate approaches 100% as the noise decreases. This is exactly the dual statement to the well known fact, called the *consistency of MDL*, that the percentage for Rissanen's MDL to identify the true model converges to 100% in the asymptotic limit of an infinite number of observations. Rissanen's MDL is regarded by many as superior to Akaike's AIC because the latter lacks this property.

At the cost of this consistency, however, the geometric MDL regards a wide range of non-degenerate models as being degenerate. This is natural, since the penalty for one degree of freedom is heavier in the geometric MDL than in the geometric AIC. Thus, the geometric AIC is more faithful to the data than the geometric MDL, which is more likely to choose a degenerate model.

4. Virtual Studio Applications

In this section, we will observe the difference in the model selection performance between the geometric AIC and the geometric MDL by doing *virtual studio* experiments: we take images of moving objects such as persons and superimpose them in a graphics-generated background [2, 15].

In order to generate a background image that is compatible to the moving viewpoint, we need to compute, at each frame in real time, the position and zooming of the camera, which arbitrarily change in the course of the shooting. A typical technique for this is to place a grid pattern colored in light and dark blue behind the object and separate the object image from the pattern image by a chromakey technique. Since the true geometry of the grid pattern is known, the position and the focal length of the camera can be determined if four or more grid points are detected in the unoccluded part of the grid pattern image [10]. However, the following two problems must be resolved:

1. When the camera optical axis is perpendicular to the pattern, the 3-D position and focal length of the camera are indeterminate because zooming out and moving the camera forward cause the same visual effect.
2. Since some unoccluded grid points become occluded while some occluded points become unoccluded as the object moves in the scene, the computed camera position fluctuates even if the camera is stationary or moving very slowly.

These problems have been dealt with by ad hoc measures in the past. We now show that they can be resolved by model selection: we model various modes of camera motion and zooming that are likely to occur in practice and choose at each frame the most appropriate one by model selection [11].

Fig. 5 shows five sampled frames from a real image sequence. Here, we used the following models (see [11] for the details of the computation): :

- The camera is stationary with fixed zooming.
- The camera rotates with fixed zooming.
- The camera linearly moves with fixed zooming.
- The camera moves arbitrarily with fixed zooming.
- The camera moves arbitrarily with linearly changing zooming.
- Everything changes.

Figs. 6(a) and 6(b) show the estimated focal lengths and the estimated camera trajectory viewed from above, respectively. Degeneracy was detected in the

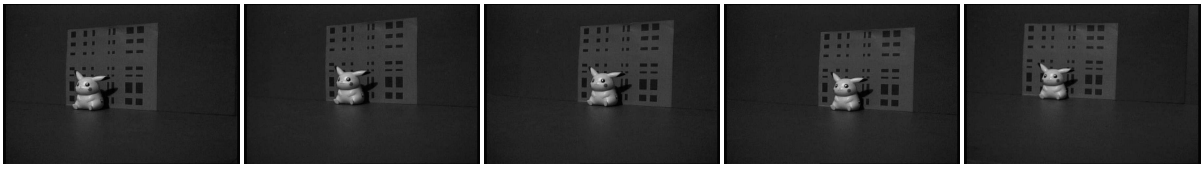
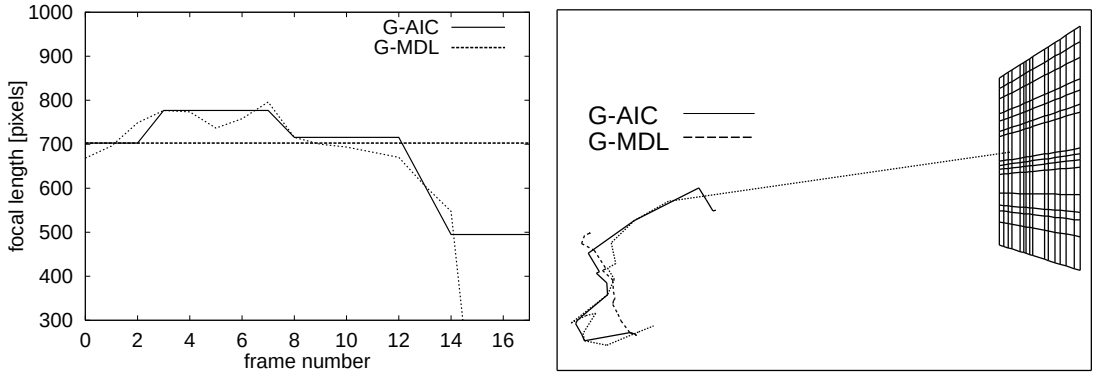


Figure 5: Sampled frames from a real image sequence.



(a) (b)

Figure 6: (a) Estimated focal lengths. (b) Estimated camera trajectory. In (a) and (b), the solid lines are for model selection by the geometric AIC; the thick dashed lines are for model selection by the geometric MDL; the thin dotted lines are for estimation without model selection.

15th frame, and the frame-wise estimation failed thereafter. We can observe that the geometric MDL tends to select a simpler model and define a more rigid trajectory than the geometric AIC, which tends select a general model and define a more flexible trajectory.

In this case, the estimation by the geometric MDL appears more consistent with the actual camera motion than the geometric AIC. But this is because we fixed the zooming and moved the camera smoothly. If we added variations to the zooming and the camera motion, the geometric MDL would still prefer a smooth motion. So, we cannot say which solution should be closer to the the true solution; it depends on what kind of solution *we expect* is desirable for the application in question.

5. Concluding Remarks

In this paper, we have defined the *geometric MDL* as the dual to *Rissanen's MDL* and given an information-theoretic interpretation. We did numerical simulations and real image virtual studio experiments and observed that the geometric AIC and the geometric MDL have different characteristics.

References

[1] H. Akaike, A new look at the statistical model identification, *IEEE Trans. Automatic Control*, **16**-6 (1974), 716–723.

[2] S. Gibbs, C. Arapis, C. Breiteneder, V. Lalioti, S. Mostafawy and J. Speier, Virtual studios: An overview, *IEEE Multimedia*, **5**-1 (1998), 24–35.

[3] H. Gu, Y. Shirai and M. Asada, MDL-based segmentation and motion modeling in a long sequence of scene with multiple independently moving objects. *IEEE Trans. Patt. Anal. Mach. Intell.* **18**-1 (1996), 58–64.

[4] K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier Science, Amsterdam, The Netherlands, 1996.

[5] K. Kanatani, Geometric information criterion for model selection, *Int. J. Comput. Vision*, **26**-3 (1998), 171–189.

[6] K. Kanatani, Model selection in statistical inference and geometric fitting, *Proc. 3rd Workshop on Information-Based Induction Sciences*, July 2000, Izu, Japan, submitted.

[7] Y. Kanazawa and K. Kanatani, Optimal line fitting and reliability evaluation, *IEICE Trans. Inf. & Syst.*, **E79-D**-9 (1996), 1317–1322.

[8] Y. Kanazawa and K. Kanatani, Optimal conic fitting and reliability evaluation, *IEICE Trans. Inf. & Syst.*, **E79-D**-9 (1996), 1323–1328.

[9] Y. G. Leclerc, Constructing simple stable descriptions for image partitioning. *Int. J. Comput. Vision* **3**-1 (1989), 73–102.

[10] C. Matsunaga and K. Kanatani, Calibration of a moving camera using a planar pattern, *Proc. 5th Symposium on Sensing via Image Information*, June 1999, Yokohama, Japan, pp. 255–260.

[11] C. Matsunaga and K. Kanatani, Calibration of a moving camera using a planar pattern: Optimal computation, reliability evaluation and stabilization by model selection, *Proc. 6th Euro. Conf. Comput. Vision*, June–July 2000, Dublin, Ireland, to appear.

[12] B. A. Maxwell, Segmentation and interpretation of multicolored objects with highlights, *Comput. Vision Image Understand.*, **77**-1 (2000), 1–24.

[13] J. Rissanen, Universal coding, information, prediction and estimation, *IEEE Trans. Inform. Theory*, **30**-4 (1984), 629–636.

[14] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific, Singapore, 1989.

[15] M. Tamir, The Orad virtual set, *Int. Broadcast Eng.*, March (1996), 16–18.

[16] P. H. S. Torr, An assessment of information criteria for motion model selection, *Proc. IEEE Conf. Comput. Vision and Pattern Recogn.*, June 1997, Puerto Rico, pp. 47–53.