グラフィクスとCAD　54－10
(1991. 12. 20)

Lie群理論による不変量の抽出：パターン認識への応用

ジェームスコール*，　村瀬　洋**，　内藤誠一郎**

Naval Research Lab.*，　NTT基礎研**

あらまし

ある画像ｘがLie群に属する変換Ｓで変形（ｘ'　＝Ｓｘ）するとしたら、この変換Ｓに対して不変となる認識関数ｆ（ｘ）を作ることが可能である。一般的にＳは多次元となる。また、複数のLie群に属する変換に対して同時に不変となる認識関数を実現することも可能である。

本報告では、そのような不変関数を実際に構成することを検討し、その際に画像の表現方法が重要となることを示す。

A Lie Group Theoretical Approach to Constructing Invariant Pattern Recognition Function

James B. Cole*, Hiroshi Murase** and Seiichiro Naito**

* Naval Research Lab.
Washington, D. C. 20375-5000 USA

** NTT Basic Research Lab.**
3-9-11 Midori-Cho Musashino-Shi

Abstract

If a set of image data, **x**, are transformed under some operation, **S**, to **x'**=**Sx**, then it is possible to construct a "recognition function," **f(x)**, which is invariant with respect to **S**, provided **S** is an element of a Lie group, which in principle could be multi-dimensional. Simultaneous invariance with respect to several Lie group transformations is also possible.

We explore the prospects for practical construction of such invariant functions and find that it is important to choose the correct image representation basis.

1. Introduction

Given an image x, we seek a "recognition function," f, such that when the image changes from $x' = Sx$, $f(x)$, $= f(x')$. That is to say we want a function f that does not change even though the image does. For example if the object of of which x is an image rotates or changes size, etc., f evaluated on the new image is still the same. We say that f is invariant with respect to x. Such an invariant function could be used for object recognition if it had different values for different objects.

To find a recognition function that is invariant with respect to S we must solve

$$f(Sx) = f(x) \tag{1}$$

for all x of interest. This is a very difficult problem that cannot in general be solved directly. If, however, the transformation S is an element of a Lie group, it is possible to construct a formal solution to Eqn. (1). This we do in Sect. 2. Detailed discussion of Lie groups can be found in Belinfante and Korman, 1972).

The way that the image is represented is very important. The representation must be such that the group properties of S are preserved. For example if S is a function of some parameter, θ, we must have that $S(-\theta)(S(-\theta)x) = x$ for any image x. Whether or not this property is preserved depends on which representation of is x used. The representation of x in turn determines the representation of S. Usually we will imagine that x is an $n \times 1$ matrix (column vector) while S is an $n \times n$ square matrix, where n is related to, but not necessarily equal to the number of pixels in the image. The components of x could be pixel brightnesses, but they could also be something else.

Finally a word about notation. To be consisent with the mathematical literature on group theory and differential geometry and also for convenience, we let x^i denote the i-th component of a column vector, while y_i denotes the i-th component of a row vector. epeated upper an lower indeces imply summation, thus $x^i y_i = \Sigma_i x^i y_i$. In addition we denote $\partial/\partial x^i$ by ∂_i.

In our approach we assume that S is an element of a one- or many- parameter Lie group (detailed discussion of Lie groups can be found in Belinfante and Korman, 1972). Instead of trying to solve (1), we seek an operator T, that acts on f, such that

$$Tf(Sx) = f(x). \tag{2}$$

T changes f in such a way as to "compensate" for the change from x to Sx . Afterwards knowing T we will attempt to solve Tf = f.

2. Formal Solution

We assume that the image data can be represented in the form $x = (x^1, x^2, ...)$, where the x^i (i = 1,2,3, ...) are "coordinates" that could represent the pixel brightnesses in the scene or which could be the coefficients in some kind of "expansion" such as that described by (Roseborough and Murase, 1990). For simplicity, we take $f(x)$ to be a real scalar function. Let $S(\theta)$ be a transformation characterized by the parameter set $\theta = (\theta^1, \theta^2, \theta^3, ...)$. S can be any operation such as translation, rotation, or dilation that forms a Lie group. An infinitesimal transformation is given by

$$S(d\theta) = \mathbf{1} + d\theta^i G_i \tag{3}$$

where the G_i are the group generators and $d\theta^i$ are infinitesimal. Throughout this paper, unless otherwise specified, we sum on upper and lower repeated indeces in accord with the Einstein summation convention. A group generator is defined as

$$\begin{aligned} G_i &= \lim_{\theta \to 0} (S(\theta) - \mathbf{1})/\theta^i \\ &= \frac{\partial}{\partial \theta^i} S(\theta)\Big|_{\theta=0} , \end{aligned} \tag{4}$$

where $S(0) = \mathbf{1}$ by definition. Applying an infinitesimal transformation, $S(d\theta)$, to x, we obtain

$$S(d\theta)\, x = x + d\theta^i G_i x . \tag{5}$$

Now making the substitutions $x \rightarrow S(d\theta)^{-1}x$ and $S \rightarrow S(d\theta)$ in (2), where $S(\theta)^{-1}$ denotes the inverse of $S(\theta)$, we find

$$Tf(x) = f(S(d\theta)^{-1}x). \tag{6}$$

For Lie groups $S(d\theta)^{-1} = S(-d\theta) = 1 - d\theta^i G_i$, thus $S(d\theta)^{-1} x = x - d\theta^i G_i x$. Substituting into (6) and expanding in a Taylor series up to the first order in $d\theta$, we obtain

$$Tf(x) = f(x) - d\theta^i (G_i x)^j \partial_j f(x) \quad , \tag{7}$$

where $(G_i x)^j$ denotes the j-th component of $(G_i x)$, and $\partial_j f(x)$ denotes the partial derivative of f with respect to x^j. T thus depends on θ and its infinitesimal form, $T(d\theta)$, is given by

$$T(d\theta) = 1 - d\theta^i (G_i x)^j \partial_j . \tag{8}$$

The generators of T, H_i, are thus

$$H_i = - (G_i x)^j \partial_j . \tag{9}$$

From the theory of Lie groups, a finite Lie group transformation can be expressed in terms of its generators by

$$T(\theta) = e^{-\theta^i H_i} . \tag{10}$$

See (Lenz, 1990) or (Kanatani, 1990) for more details of this derivation.

Defining $f^*(x, \theta) = T(\theta) f(x)$ and expanding (10) we obtain

$$f^*(x,\theta) = T(\theta)\, f(x) = e^{-\theta^i (G_i x)^j \partial_j}\, f(x)$$

$$= f(x) - \theta^i (G_i x)^j \partial_j f(x)$$

$$+ (1/2!)\, \theta^i \theta^j (G_i x)^l \partial_l((G_j x)^k \partial_k f(x)) + \cdots . \tag{11}$$

$f^*(x, \theta)$ satisfies (1) in the sense that $f^*(x,\theta) = f^*(S(\theta)^{-1}x, 0)$. Although f is arbitrary, in order to actually construct $f^*(x,\theta)$, its derivatives must be sufficiently "well behaved" that the series in (11) converges. Although $f^*(x,\theta)$ is an invariant function, it contains θ, whose value is usually not accessible from raw image data.

But knowing T, another approach is now possible. We can attempt to solve the differential equation

$$T(\theta)\, f = f \tag{12}$$

for f. This is usually a difficult problem but it is relatively easy to find a class of "first order" solutions, as we shall later see.

3. Constructing an Invariant Image Function

We now discuss how to actually construct $f^*(x,\theta)$ with respect to some transformation. For simplicity, we take $S(\theta)$ to be a one parameter Lie group. Making the replacements $G_i \rightarrow G$ and $\theta^i \rightarrow \theta$, (11) becomes

$$f^*(x,\theta) = f(x) - \theta\, (G x)^i \partial_i f(x) + (1/2!)\, \theta^2 (G x)^j \partial_j((G x)^i \partial_i f(x))$$

$$- (1/3!)\, \theta^3 (G x)^k \partial_k\big((G x)^j \partial_j((G x)^i \partial_i f(x))\big) + \cdots . \tag{13}$$

Calculating the derivatives, we obtain

$$f^*(x,\theta) = f(x) - \theta\,(Gx)^i\,\partial_i f(x) \tag{14}$$

$$+ (1/2!)\,\theta^2 \left\{(Gx)^i (Gx)^j\,\partial_j\,\partial_i f(x) + (G^2 x)^i\,\partial_i f(x)\right\}$$

$$- (1/3!)\,\theta^3 \left\{(Gx)^i (Gx)^j\,(Gx)^k\,\partial_k\,\partial_j\,\partial_i f(x)\right.$$

$$\left. + 2\,(G^2 x)^i\,(Gx)^j\,\partial_i\,\partial_j f(x) + (G^3 x)^i\,\partial_i f(x)\right\}$$

$$+ \cdots .$$

Now if we choose $f = f^{(1)}$, where $f^{(1)}$ satisfies

$$(Gx)^i\,\partial_i f^{(1)} = 0, \tag{15}$$

the second term on the right in (14) vanishes along with parts of the higher order terms. By postulating a form for $f^{(1)}$, a class of "first order" solutions to (12) can be determined. Such solutions may be useful in practical problems.

Now, to proceed further, we must determine the group generator, G.

4. Lie Group Generators for Image Data

Let us represent x as an $n \times 1$ column vector, and let $\{ e_i \}$ $(i = 1,2, \ldots, n)$ be any complete set of orthonormal basis vectors that could be used to represent the image data, such as the Walsh patterns. Expanding x in the form $x = \Sigma_i x^i e_i$, a representation for G in terms of an $n \times n$ square matrix in terms of the $\{e_i\}$ can be found. Rearranging (5), making the substitutions $x \to e_j$, $G_i \to G$ and multiplying on the left by e_i^T, we find, using the definition of G

$$\begin{aligned} e_i^T\,(S(d\theta)e_j - e_j) &= d\theta\; e_i^T\, G\, e_j \\ &\doteq d\theta\, G^i{}_j . \end{aligned} \tag{16}$$

To determine G, we apply an infinitesimal transformation, $S(d\theta)$, to e_j and expand $S(d\theta)e_j - e_j$ in the form $S(d\theta)e_j - e_j = \Sigma_i\,(de_j)^i\,e_i$, where the $(de_j)^i$ are the expansion coefficients. The elements of G are thus $G^i{}_j = (de_j)^i/d\theta$.

Defining $S(\theta)e_j = e_j(\theta)$, we can write

$$G^i{}_j = e_i{}^T \frac{d}{d\theta} e_j(0), \tag{17}$$

where $(d/d\theta)\, e_j(0)$ denotes the derivative at $\theta = 0$. G is thus determined in the form of an n X n matrix. This procedure can be carried out either "theoretically" or "empirically" using (16) or (17).

From (16), $G^i{}_j = e_i{}^T G\, e_j$. Inserting a basis vector expansion of x into $(G\,x)^i$, it is easy to show that $(G\,x)^i = x^j\, G^i{}_j$, where repeated upper and lower indeces are summed as usual. T, as given in (14), can then be expressed as

$$f^*(x, \theta) = f(x) - \theta\, x^j\, G^i{}_j\, \partial_i\, f(x) \tag{18}$$

$$+ (1/2!)\, \theta^2 \left\{ x^k\, x^l\, G^i{}_k\, G^j{}_l\, \partial_j\, \partial_i\, f(x) + x^j\, (G^2)^i{}_j\, \partial_i\, f(x) \right\}$$

$$- \cdots .$$

The foregoing developments have been carried out without reference to any specific set of basis vectors. To actually construct f^*, however, a "suitable" set must be selected, and their transformation properties defined. These choices then determine the generator representations. These choices are not completely arbitrary. Generators of such physical operations as translation and rotation must reflect their corresponding physical properties. Translation generators in x- and y- directions, G_X and G_Y, respectively, must obey $[G_X, G_Y] = 0$ (where $[a,b] = ab - ba$), since translating first in the x-direction and then in the y-direction should give the same result as performing these operations in the reverse order.

In a future paper we shall discuss representations of image data that are suitable for the analysis outline above, but we illustrate our approach with a simple example.

5. Discussion

As we have seen, the construction of an invariant recognition function, f, depends on the partial derivatives of f and on powers of the Lie group transformation generators as can be seen in Eqns. (11) and (18). It is thus very important that the series converge. Since f is not a fixed function, it is not too difficult to choose a well behaved one, but the generators depend on the image representation, and there are not so many "good" ways to represent an image.

We may wonder how humans solve the invariance problem even when the transformations are not Lie group elements. We speculate that "extra" cues, for example prior knowledge or "internal" models "fill in" the missing information.

6. Summary

Our methodology is a useful way to approach the problem of finding image invariants. The reason that it is so difficult to find an image invariant is because many transformations of image data are not Lie groups. Even when the transformations of the original objects are Lie tranformations, the transformation of the image data may not be. Futhermore even when it is, the generators may contain "large" components that cause divergence in higher order terms in the expansions (11) or (18). The question of what are appropriate representations requires further research.

References

1. Belinfante, J. G. F. and B. Kolman, B., "A Survey of Lie Groups and Lie Algebras with Applications to Computational Methods." Society for Industrial and Applied Mathematics, 1972.
2. Roseborough, J. and Murase, H., "Partial eigenvalue decomposition for very large matrices using run-length encoding." Submitted to IEEE Trans. IM.
3. Schutz, B. F. "Geometrical Methods of Mathematical Physics.", Cambridge University Press, 1985.
4. Lenz, R. "Group Theoretical Methods in Image Processing.", Springer-Verlag, 1990.
5. Kanatani, K. "Group Theoretical Methods in Image Understanding." Springer-Verlag, 1990.

盛光印刷所