DEFINITION OF NEW 3D INVARIANTS
Applications to pattern recognition problems
with neural networks

J. PRORIOL

Laboratoire de Physique Corpusculaire
de Clermont-Ferrand
CNRS/IN2P3 - Université Blaise Pascal
F-63177 AUBIERE CEDEX FRANCE

To be published in
INTERNATIONAL JOURNAL OF NEURAL SYSTEMS
DEFINITION OF NEW 3D INVARIANTS
APPLICATIONS TO PATTERN RECOGNITION PROBLEMS
WITH NEURAL NETWORKS

J. PRORIOL
Laboratoire de Physique Corpusculaire
Clermont-Ferrand (France)

ABSTRACT

We propose a definition of new 3D invariants. Usual pattern recognition methods use 2D descriptions of 3D objects, we propose a 2D approximation of the defined 3D invariants which can be used with neural networks to solve pattern recognition problems. We describe some methods to use the 2D approximants.

1-INTRODUCTION

In pattern recognition problems, some 2D invariants were defined to solve pattern recognition 2D problems\textsuperscript{1-3}. We propose in this paper, a generalization of some invariants\textsuperscript{3} to 3D problems.

This work is an extension of previous 3D invariants used to solve some high energy physics problems\textsuperscript{4}.

---

To be published in
INTERNATIONAL JOURNAL OF NEURAL SYSTEMS
In real world problems, 3D solids are described by 2D silhouettes of the solids. We propose a 2D approximation of the 3D defined invariants using the 2D silhouette data. The pattern recognition is improved by using MLP neural networks with these 2D approximate data.

After the definition of the 3D invariants and of the connected functions, we present a 2D silhouette approximation and some methods to use this approximation.

The application is done on very simple solids, we see that some simple methods can improve the classification of the solids in pattern recognition.

In the appendix 1, we present an algorithm of approximate reconstruction of 3D invariants from silhouettes data.

2-3D INVARIANTS

The ordinary polar variables in 3D are $\rho$, $\theta$, $\phi$, with the limits:

$$0 \leq \rho \leq \rho_{\text{max}}, \quad 0 \leq \theta \leq \pi, \quad 0 \leq \phi \leq 2\pi.$$ 

With the angular variables $\theta$ and $\phi$, we construct the classical orthogonal spherical functions $y_{lm}(\theta, \phi)$ \(^5\text{--}^6\).

2-1 DEVELOPMENT OF A FUNCTION

We can approximate any function $f(\rho, \theta, \phi)$ by a development:
\[ f(\rho, \vartheta, \varphi) = \sum_{nlm} a_{nlm} \mathcal{R}_{nl}(\rho) \; y_{lm}^*(\vartheta, \varphi), \]

where the parameters \(a_{nlm}\) are constants and the functions \(\mathcal{R}_{nl}(\rho)\) are orthogonal radial functions.

The limits of the integers \(n, l, m\), are : \(n = 1, \infty, 0 \leq l \leq n, -l \leq m \leq l\).

We use a function \(f\) to describe a 3D solid. If the point \(P\) of polar coordinates \(\rho, \vartheta, \varphi\), is inside the solid then we choose the value \(f = 1\), outside the solid we choose \(f = 0\).

If the polar axes are rotated with the Euler angles\(^7\) \(\alpha, \beta, \gamma\), the \(\vartheta\) and \(\varphi\) angles become \(\vartheta'\) and \(\varphi'\) angles and the new development of the \(f\) function is:

\[ f(\rho, \vartheta, \varphi) = \sum_{nlm} a'_{nlm} \mathcal{R}_{nl}(\rho) \; y_{lm}^*(\vartheta', \varphi'). \]

But we know a relation between \(\vartheta, \varphi \) and \(\vartheta', \varphi'\):\(^5-6\):

\[ y_{lm}^*(\vartheta', \varphi') = \sum_{m'} \mathcal{B}_{m'm}^{l}(\alpha, \beta, \gamma) \; y_{lm}^*(\vartheta, \varphi), \]

where the \(\mathcal{B}\) functions\(^5\) are the usual orthogonal functions with the Euler angles \(\alpha, \beta, \gamma\); the relation between the \(a_{nlm}\) and \(a'_{nlm}\) parameters is then:

\[ a_{nlm}' = \sum_{m} a_{nlm} \mathcal{B}_{m'm}^{l}(\alpha, \beta, \gamma). \]

From the orthogonal relation\(^3\):
\[ \sum_m B^j_{m*,m} (\alpha, \beta, \gamma) \ B^{j}_{m,m} (\alpha, \beta, \gamma) = \delta_{m,m}, \]

we infer the new combinations:

\[ A_{nl} = \sum_m a^{*}_{nlm} a_{nlm} \]

which are invariant for any rotation of Euler angles \( \alpha, \beta, \gamma \).

Using the properties of the \( y \) function, we can develop this function into \( P_{lm} \) associated LEGENDRE functions:

\[ y_{lm}(\theta, \varphi) = (-1)^m \sqrt{\frac{(2l+1)(l-m)!}{4\pi (l+m)!}} P_{lm}(\cos \theta) e^{imp}, \]

then the \( a_{nlm} \) coefficients are given by the relation:

\[ a_{nlm} = (-1)^m \sqrt{\frac{(2l+1)(l-m)!}{4\pi (l+m)!}} \int_{(solid)} f(\rho, \theta, \varphi) \mathcal{R}_{nl}(\rho) P_{lm}(\cos \theta) e^{imp} dV \]

where \( dV = \rho^2 d\rho d\cos \theta d\varphi \) and \( \mathcal{R}_{nl}(\rho) \) are the orthogonal functions defined in the following paragraph.

2-2 \( R_{nl} \) FUNCTIONS

The domain of variation of \( \rho \) is \( 0 \leq \rho \leq \rho_{max} \). The \( R_{nl} \) radial functions are orthogonal according to the relation:

\[ \int R_{n'l'}(\rho) R_{nl}(\rho) \rho^2 d\rho = \delta_{n'n} \delta_{l'l} \]

where \( \delta \) is the usual Dirac function.
The changing of variable \( x = 2 \rho / \rho_{\text{max}} - 1 \) gives a new domain of variations for this variable: \(-1 \leq x \leq 1\). The \( R_{nl} \) functions are defined using the JACOBI polynomials\(^8\) \( p_n^{(\alpha, \beta)}(x) \); a \( R_{nl} \) normalized value is:

\[
R_{nl}(\rho) = \sqrt{\frac{2n+3}{2} \frac{1-x}{(1+x)^{l+1}}} p_n^{(0,2l+2)}(x).
\]

3-APPROXIMATE SILHOUETTE INVARIANT

Looking at a solid from one direction, we cannot get all the informations on 3D solids; we get only 2D informations which describe the silhouettes of the solid.

We can approximate the \( a_{nlm} \) coefficients using the \( \theta \) value \( \pi/2 \). We set:

\[
b_{nlm} = (-)^m \frac{(2l+1)(l-m)!}{4\pi(l+m)!} \int p_{lm}(0) \int_{(silh)} g(\rho, \phi) R_{nl}(\rho) e^{im\phi} \rho d\rho d\phi
\]

where the \( g \) function defines the silhouette in the \( xy \) plane.

If the point \( P \) of polar coordinates \( \rho, \phi \) is inside the silhouette, we use \( g = 1 \); outside the silhouette, we use \( g = 0 \).

With the \( b_{nlm} \) coefficients, we compute the \( B_{nl} \) approximate invariants similar to the \( A_{nl} \) ones.

4-PATTERN RECOGNITION

In real world problems, the pattern recognition of 3D solids is done usually with 2D data. These data come from the
silhouettes of the solids.

The invariants \( \text{B}_{n1} \), while using the silhouettes, are computed with the approximate formula of paragraph 3. But approximate invariants and the noise of the data do not allow a simple classification of several similar solids.

A neural network usually gives a good classification with a suitable set of approximate 2D training data. To perform this classification, we can use two methods.

The first method is to use as input data, the approximate invariants from a set of one, two or several simultaneous silhouettes of the solids viewed from one or several directions.

The second method is to perform an approximate reconstruction of one solid from several silhouettes. From this reconstructed solid we compute the \( \text{A}_{n1} \) invariants. In appendix 1, we present an algorithm used in our computations. The reconstruction is not exact and is far from being perfect.

To test these methods, we have generated data using five simple solids without noise. The thickness of the different solids is the same. The projections of the solids on a plane are: a square (solid n° 1), an equilateral triangle (n° 2), a rectangle (n° 3), an isosceles triangle (n° 4), a cercle (n° 5). The silhouettes of these solids are 2D
pictures of 50*50 pixels. Some views, with random Euler angles, are presented in figure 1.

We applied the algorithm of appendix 1 to reconstruct the five solids and compute the $A_{nl}$ parameters. In table 1, we give different $A_{00}$ values for different reconstructions. The first column gives the numbering of the solids. In the second column, we give the exact $A_{00}$ value from the exact formula of paragraph 2-1. In the other columns, we give the reconstructed $A_{00}$ values after the application of the algorithm of appendix 1 on n silhouettes of the solids. The inputs of this appendix are the n silhouettes of one solid viewed from several directions. In our computation, the directions are regularly placed around the solid in an equatorial plane. For a large number of views, the reconstruction is rather fair:

<table>
<thead>
<tr>
<th>solid</th>
<th>exact</th>
<th>15 views</th>
<th>7 views</th>
<th>5 views</th>
<th>3 views</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.19555</td>
<td>0.16756</td>
<td>0.23995</td>
<td>0.30679</td>
<td>0.20967</td>
</tr>
<tr>
<td>2</td>
<td>0.08703</td>
<td>0.08783</td>
<td>0.10771</td>
<td>0.12152</td>
<td>0.17077</td>
</tr>
<tr>
<td>3</td>
<td>0.02121</td>
<td>0.01867</td>
<td>0.02165</td>
<td>0.02460</td>
<td>0.03921</td>
</tr>
<tr>
<td>4</td>
<td>0.00897</td>
<td>0.00722</td>
<td>0.00917</td>
<td>0.01089</td>
<td>0.01491</td>
</tr>
<tr>
<td>5</td>
<td>0.12365</td>
<td>0.10406</td>
<td>0.15424</td>
<td>0.17662</td>
<td>0.22261</td>
</tr>
</tbody>
</table>

Table 1: Reconstruction of the invariant $A_{00}$ with different number of 2D views of the solids.
5-PATTERN RECOGNITION ON 5 SOLIDS

We present three methods of pattern recognition on the 5 solids defined in the paragraph 4. For each method, we generate 2 sets of random silhouettes data for the 5 solids. A set is used for the training of a neural network, the other set is used for the validation of the training and the test.

The neural network used is a four layers perceptron (MLP) trained with back-propagation. The architecture is: m-h₁-h₂-c, with m input variables, h₁ and h₂ neurons for the hidden layers and c output neurons for the c classes of the classification. The choice of the number of neurons is such as: m ≥ h₁ > h₂ > c. This kind of architecture gives good classification results with difficult high energy physics problems.

The 21 $B_{nl}$ invariants used in this study are given by the limits of the n and l parameters: $0 ≤ n ≤ 5$, $0 ≤ l ≤ n$

5-1 ONE SILHOUETTE

The first method is the one silhouette method. The observation of the solid is done from one direction.

We have generated 2 sets of 5*200 random silhouettes data of the solids. The MLP neural network is a 21-21-10-5 one. During the training, we use a training set for the back-propagation and a validation set similar to the training set.
We compute the costs on the two sets. The training is stopped when the cost of the validation set begins to increase.

After the training of the MLP, we get a classification matrix from the test set. This matrix is given in table 2. In this paper we present several similar matrices, the interpretation of the different numbers of these tables is given in appendix 2.

<table>
<thead>
<tr>
<th>origin</th>
<th>classification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>192</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2: Classification matrix using the silhouette method to compute the $B_{nl}$ parameters. We use one silhouette. The percentage of well classified silhouettes is 93.2%.

5-2 TWO SILHOUETTES

The two solids are observed from two directions. The two silhouettes are generated with an angle $\alpha$ between the two axis of observation. We have generated two sets of 5*200*2 silhouettes data. We use the invariants from the two silhouettes. The MLP is then a 42-21-10-5 one.

We have got the classification matrix for two $\alpha$ angles. In table 3 we consider the case $\alpha=30^\circ$:
Table 3: Classification matrix using the silhouette method to compute the $B_n^1$ parameters. We use two silhouettes data. The angle between the two directions of observation is $\alpha=30^\circ$. The percentage of well classified silhouettes is 96.3%.

In table 4, we consider the case $\alpha=90^\circ$.

Table 4: Classification matrix using the silhouette method to compute the $B_n^1$ parameters. We use two silhouettes data. The angle of between the two axis of observation is $\alpha=90^\circ$. The percentage of well classified silhouettes is 99.2%.

5-3 3D RECONSTRUCTION

In the general case, the solids are observed from $n$ directions. In the present reconstruction, the solids are observed from two directions. The angle between the two axis of observation is $90^\circ$. 
The reconstruction is done using the algorithm given in the appendix 1.

We have reconstructed two sets of $5\times100$ solids from $5\times100\times2$ silhouettes. We have computed 21 $A_{nl}$ invariants and used a MLP 21-21-10-5.

The classification matrix is given in table 5.

<table>
<thead>
<tr>
<th>origin</th>
<th>classification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>98</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 5: Classification matrix using the reconstruction method to compute the $A_{nl}$ parameters. The reconstruction is done from two silhouettes. The angle between the two axis of observation is 90°. The percentage of well classified silhouettes is 98.6%.

6-CONCLUSION

We have defined new 3D invariants. When it is possible to compute these invariants, the classification of patterns is easy.
But the problem is more difficult when we dispose only of 2D informations on 3D solids. We have defined approximate invariants. The utilisation of neural networks using the approximate invariants as input data, gives a good classification of patterns.

In the proposed applications, we use very simple solids, and we show that several silhouettes and an approximate reconstruction of the invariants of the solids improve the classification.
7- REFERENCES

1. T.H. REISS "Recognizing planar objects using invariant image features"
   Springer-Verlag Heisenberg (1993)

   in Proceedings ICANN'95 Ed F. FOGELMAN-SOULIE
   EC2 PARIS (1995)

3. S.J. PERANTONIS, P.J.G. LISBOA

4. J. PRORIOL
   Int journal of neural systems 3 (supp 1992) 271

5. A.R. EDMONDS "Angular momentum in quantum mechanics"
   Princeton University Press Princeton (1957)

6. J.M. NORMAND "Rotation in quantum mechanics"
   North-Holland AMSTERDAM (1980)

7. H. GOLDSTEIN "Classical mechanics"
   Addison-Wesley Cambridge (1956)

8. W. MAGNUS, F. OBERHETTINGER, R.P. SONI
   "Formulas and theorems for the special functions of mathematical physics"
   Springer-Verlag Heidelberg (1966)
APPENDIX 1 Reconstruction of the solid

Inputs:
The number of observations is NFACES,
the \( \nu \) angles of the observations are \( c_i \), \( i=1..NFACES \),
the number of 3D points is \( NMAX*NMAX*NMAX \),
the silhouette is given by \( nf(i2,i,j) \), \( i2=1..NFACES \)
ca and ba are logical variables,
the function \( f=1 \) is written \( n(i,j,k)=1 \),
\( dx=2/NMAX \).

\[
\text{do } i=0,NMAX \\
\quad \text{do } j=0,NMAX \\
\quad \quad \text{do } k=0,NMAX \\
\quad \quad \quad n(i,j,k)=0 \\
\quad \quad \text{enddo} \\
\quad \text{enddo} \\
\text{enddo}
\]

\[
\text{do } i=0,NMAX \\
\quad x=-1+i*\text{dx} \\
\quad \text{do } j=0,NMAX \\
\quad \quad y=-1+j*\text{dx} \\
\quad \quad \text{do } k=0,NMAX \\
\quad \quad \quad z=-1+k*\text{dx} \\
\quad \quad \quad \text{do } i2=1,NFACES \\
\quad \quad \quad \quad x1=x*\cos(i2)-z*\sin(i2) \\
\quad \quad \quad \quad i3=\text{int}(x1+1)/\text{dx} \\
\quad \quad \quad \quad ca(i2)=(nf(i2,i3,k).eq.1) \\
\quad \quad \quad \text{enddo} \\
\quad \quad \quad \quad \text{ba=(ca(1).and.ca(2))} \\
\quad \quad \quad \text{do } i2=3,NFACES \\
\quad \quad \quad \quad \text{ba=(ba.and.ca(i2))} \\
\quad \quad \quad \text{enddo} \\
\quad \quad \text{if (ba) then} \\
\quad \quad \quad n(i,j,k)=1 \\
\quad \quad \text{endif} \\
\quad \text{enddo} \\
\text{enddo}
\]
APPENDIX 2 Classification matrix

A classification matrix from a MLP classifier applied on a test set of silhouettes is given by the following table:

<table>
<thead>
<tr>
<th>original class</th>
<th>class of classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>c11 c12 c13 c14 c15</td>
</tr>
<tr>
<td>2</td>
<td>c21 c22 c23 c24 c25</td>
</tr>
<tr>
<td>3</td>
<td>c31 c32 c33 c34 c35</td>
</tr>
<tr>
<td>4</td>
<td>c41 c42 c43 c44 c45</td>
</tr>
<tr>
<td>5</td>
<td>c51 c52 c53 c54 c55</td>
</tr>
</tbody>
</table>

The MLP classifier takes a silhouette of the test set. The original class of the solid is 1 to 5. The classifier attributes at the silhouette a class from 1 to 5: it is the classification of the silhouette by the classifier.

During a work of classification, we present all the silhouettes of the test set to the classifier. The $c_{ij}$ number is the number of silhouettes of original class $i$ classified into the class $j$. When the classifier is perfect, we get $c_{ij}=0$ if $i$ is different of $j$.

The total number of silhouettes classified by the classifier is: $N = \sum \sum c_{ij}$.

The sum $D_i=\sum_j c_{ij}$ is the number of silhouettes of the original class $i$. 
The sum $E_j = \sum_i c_{ij}$ is the number of silhouettes classified into the class $j$.

We state that a silhouette is well classified if the class of the classification is identical to the original class.

Then the percentage of well classified silhouettes by the classifier is: $r = \sum_i c_{ii} / N$. 
FIGURE 1
Three silhouettes (SIL1,SIL2,SIL3) obtained with random Euler angles $\alpha$, $\beta$, $\gamma$ for the five solids.