Multi-view Object Representation with Inverse Difference Pyramid Decomposition

ROUMEN KOUNTCHEV
Radio-communications Dept.
Technical University of Sofia
Bul. Kl. Ohridsky, 8
Sofia 1000, BULGARIA
rkountch@tu-sofia.bg
http://www.tu-sofia.bg

VLADIMIR TODOROV
T&K Engineering Co.
Mladost 3, POB 12
Sofia 1712
BULGARIA
todorov_vl@yahoo.com

ROUMIANA KOUNTCHEVA
T&K Engineering Co.
Mladost 3, POB 12
Sofia 1712
BULGARIA
kountcheva_r@yahoo.com

Abstract: In the paper is presented one new method for multi-view object representation based on image decomposition with Inverse Difference Pyramid. The method permits to obtain a very efficient description of the multi-view images using one of them as a reference one. The decomposition has a relatively low computational complexity because it is based on orthogonal transforms (Walsh-Hadamard, DCT, etc.). The relations which exist between transform coefficients from the consecutive decomposition layers permit significant reduction of the coefficients needed for the high-quality object representation.

Key-Words: Multi-view images, multi-view representation, pyramid decomposition

1 Introduction
Scientists and industry increasingly need multi-view representations of objects in the built environment and the demand for such kind of information is ever increasing. Some of the typical application areas are:

- 3D geographical information systems (GIS);
- hazardous and accident site survey;
- quality control for production lines;
- facility or construction management;
- object data mining, etc.

Two different types of image features can be extracted: those that are directly related to the 3-D shape of the part of the object being viewed and features, that result from the 3-D to 2-D down projection (the second one can be ambiguous because part of the 3-D shape information is lost during the projection). The essence of the recognition problem is to relate the structures found in the image with the underlying object models. Other important issues involved in structural recognition are:

- adequacy of the representation for the kind of objects encountered;
- selection and extraction of visual primitives;
- description of the spatial relations between primitives;
- matching image structures to models;
- inference of structural object descriptions from examples.

The selection of suitable shape primitives is of central importance. For efficient recognition, they should be *expressive* in the sense that a combination of only a few of them, or even a single one, can facilitate object identification. On the other hand, the available shape primitives should be *general* enough to model a large range of object categories. They should be detectable from images reliably in a bottom-up fashion and should be *non-incidental*, in the sense that they are unlikely to occur from random configurations in space [1, 2].

The pyramidal image representation is one of the frequently used techniques. The object reconstruction at a given pyramid level is based on the feature-based matching approach. The first step required at each level is the extraction of salient features (points and/or lines) together with their topological relations, which is a process controlled by a model of what is expected to be found in the images. Having detected features in two or more images, the correspondence problem has to be solved. The general approach seeks correspondences in object space, because this approach is more flexible with regard to handling occlusions and surface discontinuities. The task-dependent local model of the object surface is then provided, and false correspondences are detected from bad fits to that model in object space [3, 4].

The Inverse Difference Pyramid (IDP) decomposition suits the peculiarities of this basic
The IDP-based object representation (and correspondingly - the salient features extraction) is performed in the spectrum domain. The creation of consecutive approximating images with increasing quality suits very well the widely used algorithms for image data mining [5]. Together with this the IDP decomposition offers specific advantages when the creation of 3D object model is concerned.

The paper is arranged as follows: in Section 2 are given the principles of the multi-view image representation with 2-layer Inverse Difference Pyramid (IDP) Decomposition; in Section 3 are included some experimental results and Section 4 is the Conclusion.

## 2 Multi-view object representation based on 2-layer IDP

The IDP–based approach for multi-view representation is similar with the human way for object analysis and recognition [2] – starting with low similarity and continuing with better resemblance. By analogy, the object model creation starts with the coarse approximation, which corresponds to the lower pyramid layer, and continues with better approximation obtained in the next one(s). The object model is completed when the required quality for the best approximation is obtained.

The multi-view image representation is obtained by processing the object views taken from various positions. The selection of the view point depends on the application. The general approach is the view points to be placed in a sphere, surrounding the object (i.e. the object is in the center of the sphere). For some applications, the view points could be arranged in a line, or in a circle. An example multi-view arrangement in a part of a sphere is shown in Fig. 1 and arrangement in a circle – in Fig. 2. The optimum number of view points (correspondingly – the angles between them) depends on the application as well. One of the views is always used as a reference one.

The algorithm for the multi-view object representation based on 2-layer IDP is presented below.

The matrix of the processed image is initially divided into sub-images of size \( r \times r \) pixels (usually, 8x8 or 4x4 pixels each). The following main assumptions are accepted:

- The number of the object’s multi-view images is \((2N+1)\);
- The matrices \([B_n]\) of the multiple views for \( n=R, \pm 1, \pm 2, \ldots, \pm N \) are of size \( r \times r \) pixels.

- The reference image \( R \), represented by the matrix \([BR]\) is chosen to be placed in the middle of the sequence of multi-view images \([B_n]\) placed in a line (or in the center of the part of the view sphere).

The processing comprises the following steps:

1. For the lower decomposition layer \( p=0 \) is calculated the transform \([S_0^R]\) of the reference image \([BR]\) using direct orthogonal transform:

\[
[S_0^R][T_0][B_R][T_0],
\]

where \([T_0]\) is the matrix for direct orthogonal transform of size \( r \times r \) (same as the size of the sub-image).

![Fig. 1. Example layered multi-view arrangement in parallel circles, which build a part of a sphere around the object.](image1)

![Fig. 2. View points arrangement in a circle](image2)

2. The approximating transform for the reference image is calculated:

\[
[S_0^R] = [m(u, v) s_0^R (u, v)],
\]
where \( m_0(u,v) \) is an element of the matrix-mask, which defines the retained coefficients used for the coarse approximation:

\[
m_0(u,v) = \begin{cases} 
1, & \text{if } s^R_0(u,v) - \text{retained coefficient,} \\
0, & \text{in all other cases,}
\end{cases}
\]  

(3)

In result is obtained the coarse approximation of the processed image. The number of retained transform coefficients, defined by the matrix-mask influences the quality of the obtained approximation (more coefficients ensure higher quality, but lower efficiency of the object description).

3. The approximated reference image \( \hat{B}_0^R \) is calculated, using inverse orthogonal transform in correspondence with the relation:

\[
[\hat{B}_0^R] = [T_0']^{-1}[\hat{S}_0^R][T_0],
\]  

(4)

where \( [T_0'] = [T_0]^{-1} \) is the matrix of the inverse orthogonal transform, of size \( r \times r \).

4. The difference matrix is calculated in accordance with the relation:

\[
[E_n] = [B_n] - [\hat{B}_0^R].
\]  

(5)

5. The difference matrix is divided into 4 sub-matrices:

\[
[E_n] = \begin{bmatrix} [E^1_n] & [E^2_n] \\
[E^3_n] & [E^4_n] \end{bmatrix},
\]  

(6)

where \( [E^i_n] \) for \( i=1,2,3,4 \) are sub-matrices of size \( (r/2) \times (r/2) \) each.

6. For the next (higher) decomposition layer \( p=1 \) is calculated the transform \( [S^R_n] \) of the submatrix \( i \) of the difference \( [E_n] \), using direct orthogonal transform, as follows:

\[
[S^R_n] = [T_i][E^i_n][T_i] \quad \text{for } i=1,2,3,4,
\]  

(7)

where \( [T_i] \) is the matrix for direct orthogonal transform of size \( (r/2) \times (r/2) \).

7. The approximating \( i^{th} \) transform is calculated:

\[
[\hat{S}^R_{i,j}] = [m_j(u,v) s^R_{i,j}(u,v)],
\]  

(8)

where \( m_j(u,v) \) is an element of the matrix-mask, which defines the retained spectrum coefficients:

\[
m_j(u,v) = \begin{cases} 
1, & \text{if } s^R_j(u,v) - \text{retained coefficient,} \\
0, & \text{in all other cases}
\end{cases}
\]  

(9)

The retained coefficients in the second decomposition layer are usually different from these in the initial layer and their number depends on the restored image quality, needed for the application.

8. For the decomposition layer \( p=1 \) of every view sub-image \( [B_n] \) is calculated the difference:

\[
[E_n] = [B_n] - [\hat{B}^R_n] \quad \text{for } n=R,±1,±2,±3,±N
\]  

(10)

9. The difference matrix is divided into 4 sub-matrices:

\[
[E_n] = \begin{bmatrix} [E^1_n] & [E^2_n] \\
[E^3_n] & [E^4_n] \end{bmatrix}.
\]  

(11)

where \( [E^i_n] \) for \( i=1,2,3,4 \) are sub-matrices of size \( (r/2) \times (r/2) \).

10. The \( i^{th} \) transform \( [S^R_n] \) of the sub-matrix of the difference \( [E_n] \) is calculated, using direct orthogonal transform:

\[
[S^R_n] = [T_i][E^i_n][T_i] \quad \text{for } i=1,2,3,4.
\]  

(12)

11. The approximated \( i^{th} \) transform is calculated:

\[
[\hat{S}^R_{i,j}] = [m_j(u,v) s^R_{i,j}(u,v)],
\]  

(13)

where \( m_j(u,v) \) is an element of the matrix-mask for the layer \( p=1 \).

12. The coefficients of the matrices \( [\hat{S}^R_n] \) and \( [\hat{S}^R_{i,j}] \) are coded losslessly for \( i=1,2,3,4 \) and \( n=±1,±2,±3,±N \) from IDP layers \( p=0,1 \) and arranged in a common massif. The block diagram of the IDP decomposition for the reference image is shown in Fig. 3.a, and the block diagram of the coder for multi-view object representation based on 2-layer IDP is shown on Fig. 3.b. The decoding is performed in reverse order (Fig. 4).
4 Experimental results

For higher efficiency the approach presented here is based on the use of a fixed set of transform coefficients (these of lowest spatial frequency). For the low decomposition layers a set of 4 coefficients is usually enough. In the last (highest) layer is possible to use one coefficient only, which results in better description efficiency. For the experiments was used a 2-layer decomposition in which only 1 coefficient was retained for the higher layer. For the experiments was used IDP with the Walsh-Hadamard orthogonal transform (WHT). The views were obtained by moving the photo camera in a line, with an angle of 4° between every two adjoining view positions. The total number of views in a line was 11. The reference image was chosen to be the one in the middle of the sequence. Two more view lines (11 views each) were arranged by moving the photo camera 4° up and down in correspondence to the first. The processed images were of size 864×576 pixels, 24 bpp each. The reference image from one of the test groups is shown in Fig. 5.

For the experiments the basic sub-image in the low decomposition layer was 8 × 8 pixels and the number of the low-frequency transform coefficients was 4. The size of the coarse approximation file (layer 1) for the reference view was 15418 B and the corresponding PSNR was 37.83 dB. The mean PSNR for the whole group of 11 views for 2-layer IDP was 36.32 dB.

\[
CR = \frac{(2N + 1)r^2 b_0}{(L_0 + 2NL_1)b_2},
\]

where \(b_0\) and \(b_2\) represent the number of bits for one pixel and one transform coefficient correspondingly; \(L_0\) and \(L_1\) – the number of the retained coefficients for the IDP layers \(p=0\) and \(p=1\). The so defined compression ratio does not represent the influence of the lossless coding of the coefficients’ values performed for IDP layers \(p=0\) and \(p=1\).

In the column “L2 file size” is given the size of the corresponding approximations for the higher decomposition layer in Bytes. The compression ratio (CR) was calculated for the whole group of images, i.e. the total data needed for the representation of all 11 views was compared with the uncompressed data for the same images. In the column named “CR Layer 2” is given the compression ratio obtained for the corresponding representations of the decomposition layer 2 only.

Similar investigation was performed for another 11 views of the same objects, placed in a line positioned at 4° higher than the first one. The angles between adjacent views were 4°. In this case the reference view was chosen to be at the end of the sequence (next to View No. 10). The results are given in Table 2.

<table>
<thead>
<tr>
<th>View No.</th>
<th>CR Layer 2</th>
<th>L2 file size [B]</th>
<th>PSNR L2 [dB]</th>
<th>CR (group)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ref.</td>
<td>90.62</td>
<td>16 746</td>
<td>35.65</td>
<td>129.98</td>
</tr>
<tr>
<td>1</td>
<td>70.80</td>
<td>21 088</td>
<td>34.47</td>
<td>126.32</td>
</tr>
<tr>
<td>2</td>
<td>75.00</td>
<td>19 906</td>
<td>35.56</td>
<td>123.99</td>
</tr>
<tr>
<td>3</td>
<td>80.75</td>
<td>18 490</td>
<td>34.53</td>
<td>122.74</td>
</tr>
<tr>
<td>4</td>
<td>82.23</td>
<td>18 157</td>
<td>34.52</td>
<td>121.83</td>
</tr>
<tr>
<td>5</td>
<td>86.17</td>
<td>17 326</td>
<td>34.53</td>
<td>121.40</td>
</tr>
<tr>
<td>6</td>
<td>90.26</td>
<td>16 541</td>
<td>34.61</td>
<td>121.34</td>
</tr>
<tr>
<td>7</td>
<td>89.36</td>
<td>16 708</td>
<td>34.70</td>
<td>121.23</td>
</tr>
<tr>
<td>8</td>
<td>88.04</td>
<td>16 959</td>
<td>34.81</td>
<td>121.03</td>
</tr>
<tr>
<td>9</td>
<td>86.99</td>
<td>17 162</td>
<td>35.11</td>
<td>120.78</td>
</tr>
<tr>
<td>10</td>
<td>85.16</td>
<td>17 532</td>
<td>35.26</td>
<td>120.43</td>
</tr>
</tbody>
</table>

Mean PSNR = 34.89 dB

The results obtained are close to those given in Table 1, but the CR and the PSNR are a little lower,
because the reference view for the second line was set to be this at the end of the sequence and as a result, the correlation between the consecutive views is lower.

Additional test was performed for a line of consecutive views positioned at 4° down in respect of the first one. The global results are as follows: the PSNR for the whole group (3 lines of views) is 34.8dB and the compression ratio is CR = 120.1. This means that for the group of 33 color images (one reference image and 32 views arranged in 3 lines) each of size 864 × 576 pixels, was achieved a compression ratio CR > 120. The quality of the views was visually lossless, because the errors in images which have a PSNR higher than 32 dB are imperceptible.

The tests performed simulated a matrix of 33 views arranged in a rectangle of size 11 × 3. Best results are obtained when the reference view is placed in the center of the viewing matrix.

The main advantage of the new approach is that it ensured high compression and very good quality of the visual information. In spite of the global approach when the data storage is concerned, each view could be restored individually and used.

5 Conclusion
In the paper is presented one new method for 3D object representation based on pyramidal image decomposition. The method ensures very efficient description of the multi-view images by using one of them as a reference one. The decomposition has a relatively low computational complexity because it is based on orthogonal transforms (DCT, Walsh-Hadamard, etc.). For example, the computational complexity of decompositions, based on wavelet transforms is much higher. In the examples was used the WH transform, but DCT or some other transforms are suitable as well. The relations existing between transform coefficients from the consecutive decomposition layers permit significant reduction of the coefficients needed for the high-quality object representation [6]. The number of the necessary views depends on the application. For example, the view area could be restricted to some angle or scale, etc.

The experimental results proved the ability to create efficient multi-view object representation based on the IDP decomposition. The task is easier when the image of a single object has to be represented. In the examples, presented here, two convex objects were represented and they should be searched together. The significant compression of the data representing the multiple views ensures efficient data storage and together with this - fast access and search in large image databases.

The IDP representation is suitable for tasks requiring the analysis of complicated scenes (several objects searched together or context-based search). This is possible, because the lowest layer of the pyramidal decomposition consist of sub-images, processed individually. In result, more than one object (described individually) could be searched together.

Additional advantage is the similarity of the transform coefficients from any two adjacent decomposition layers, which is a basis for the creation of flexible algorithms for the transformation of the already created object representation into higher or lower scale.

Acknowledgement
This work was supported by the National Fund for Scientific Research of the Bulgarian Ministry of Education and Science, Contract VU-I 305.

References:
Coded multi-view data

\[ \hat{s}_i^R(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^N(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^R(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^N(u,v), i = 1,2,3,4 \]

\[ \Sigma \]

\[ \Sigma \]

\[ [E_{R}] \]

\[ [E_{N}] \]

\[ [B_{R}] \]

\[ [B_{N}] \]

View +N

View 1

Ref. view

View -N

p = 1

p = 0

Coded multi-view data/decoding

\[ \hat{s}_i^N(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^R(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^N(u,v), i = 1,2,3,4 \]

\[ \hat{s}_i^R(u,v), i = 1,2,3,4 \]

\[ \Sigma \]

\[ \Sigma \]

\[ [E_{R}] \]

\[ [E_{N}] \]

\[ [B_{R}] \]

\[ [B_{N}] \]

View +N

View 1

Ref. view

View -N

Input

\[ \Sigma \]

\[ \Sigma \]

\[ [E_{R}] \]

\[ [E_{N}] \]

\[ [B_{R}] \]

\[ [B_{N}] \]

Fig. 3.b. Block diagram of the coder for multi-view object representation based on 2-layer IDP.

Fig. 4. Block diagram of the decoder