Aggregation on the Basis of Structure Recognition

Jagdish Lal Raheja,
Department of Cartography
Technical University of Munich

Structure Recognition

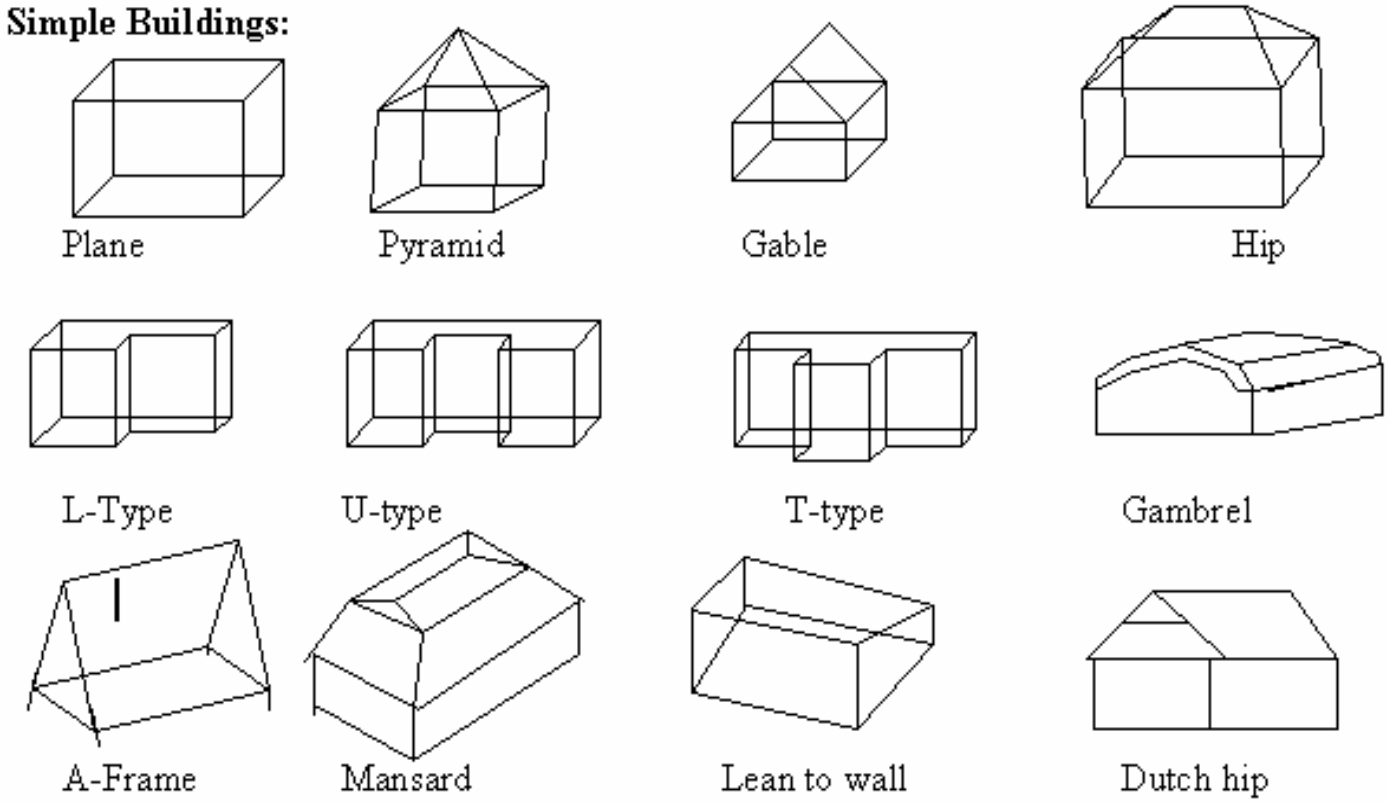
Structure recognition: identify specific cartographic objects or aggregates as well as spatial relations, and measures of importance

A city model consists mainly of buildings of differnet shape, style and roads.

Topics to be discussed

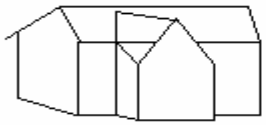
- Identification of objects (buildings).
- Identification of spatial relations.
- Aggregations rules based upon sturcture recogoniton.

Simple Buildings

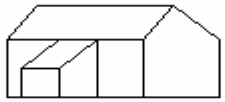


Complex buildings

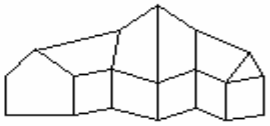
a. buildings which has a roof as a combination of same or different roof styles.



Cross gable

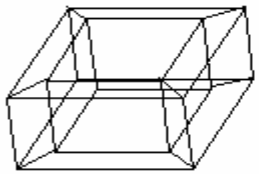


Gable – Lean

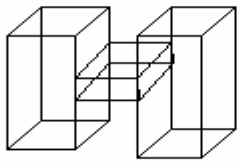


Arrow gable

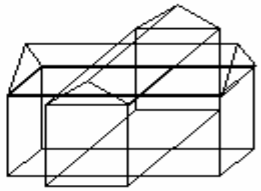
b. building as a group of small buildings joined together forming a special shape



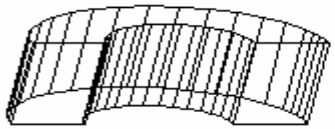
Closed square



bridge

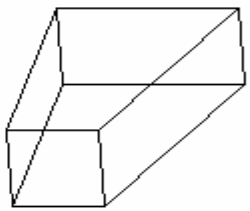


cross



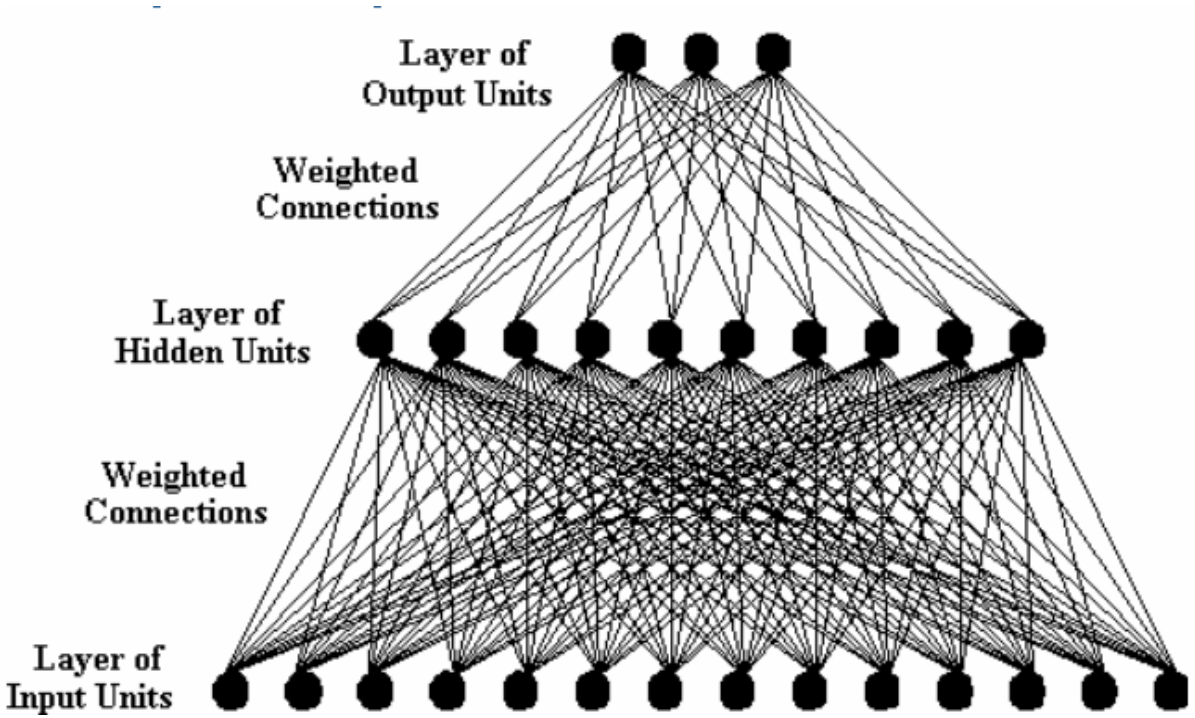
Arc

Trapezium

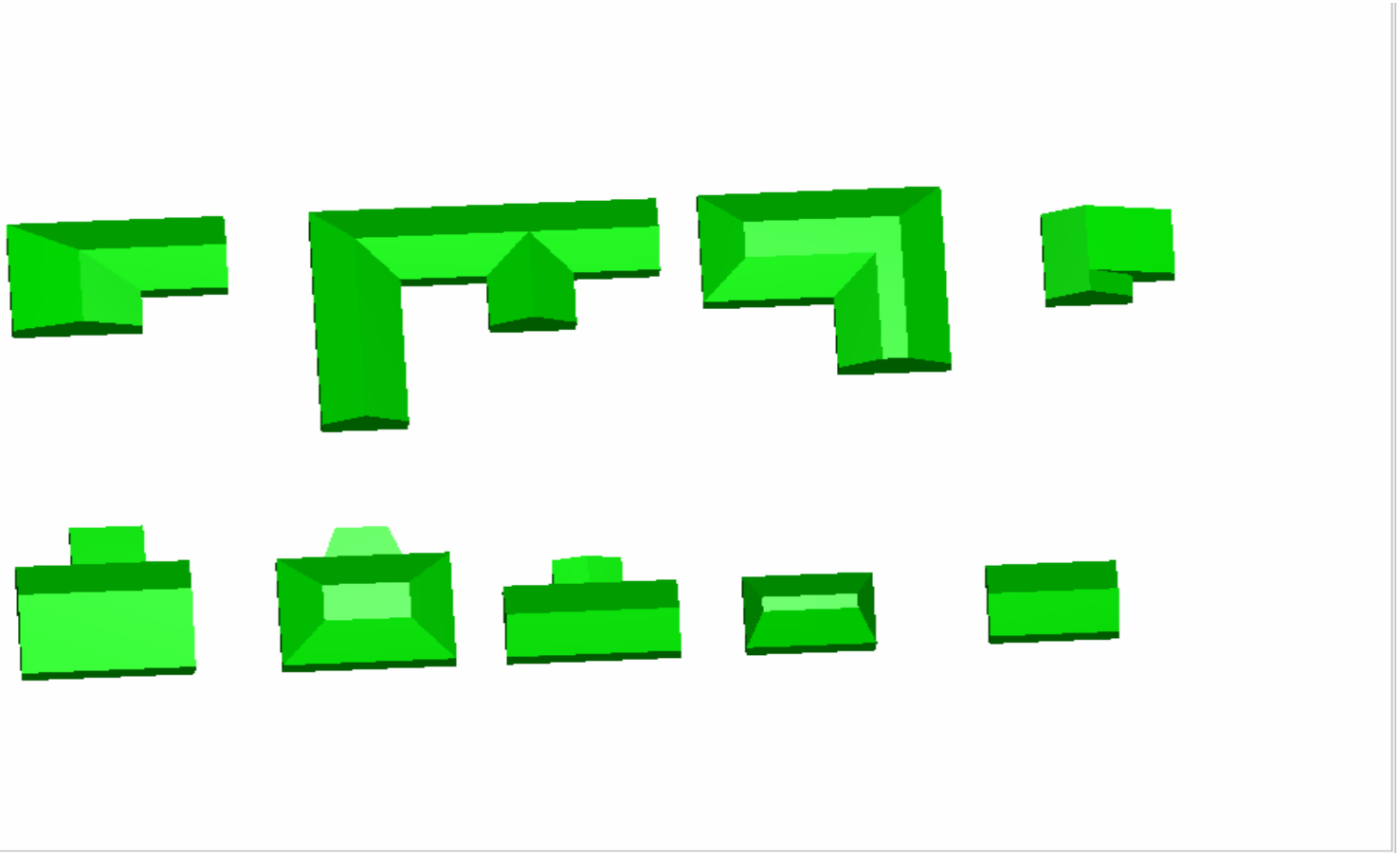


Building recognition using NN

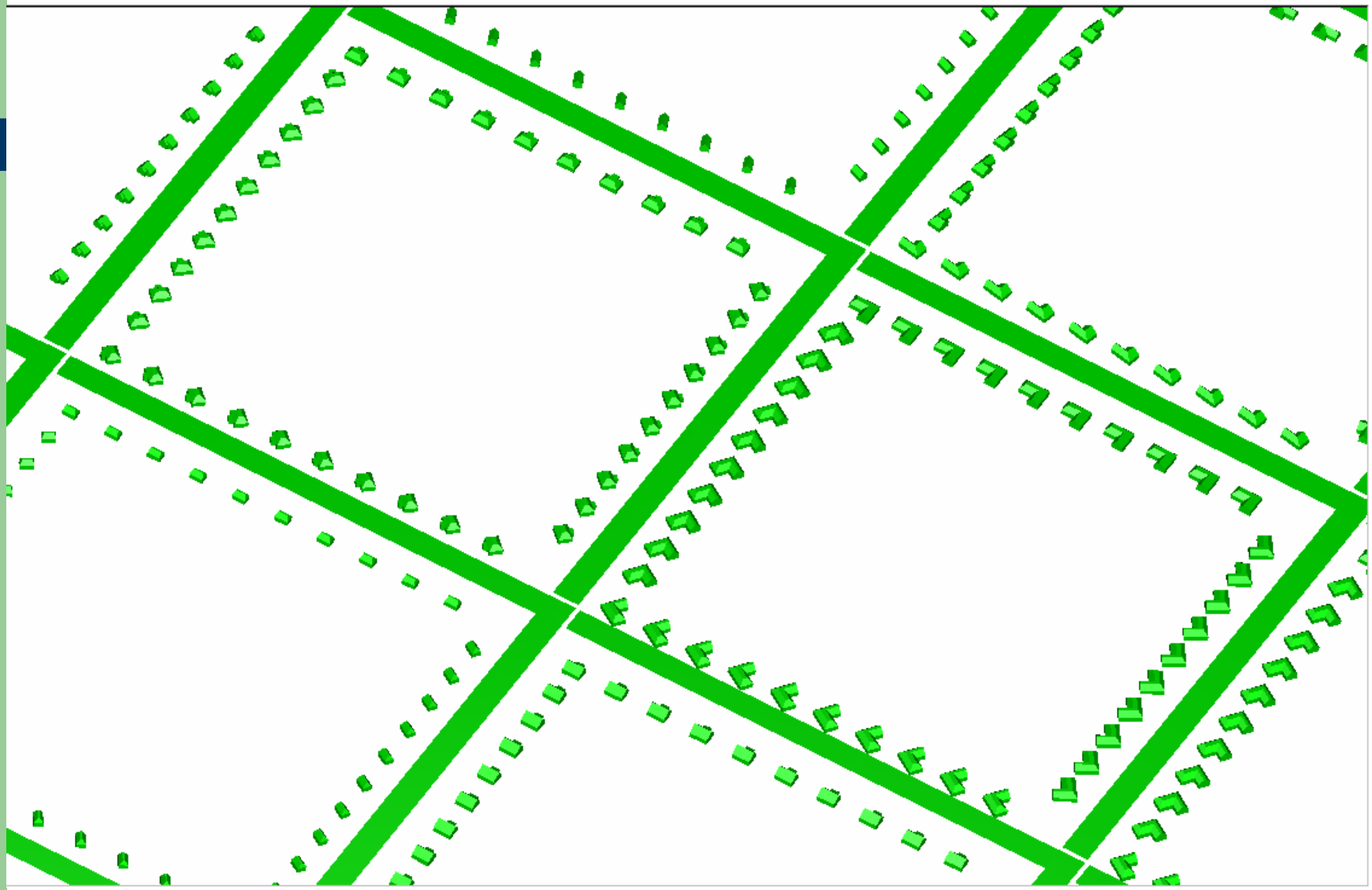
NN-bsiac structure



Test Data



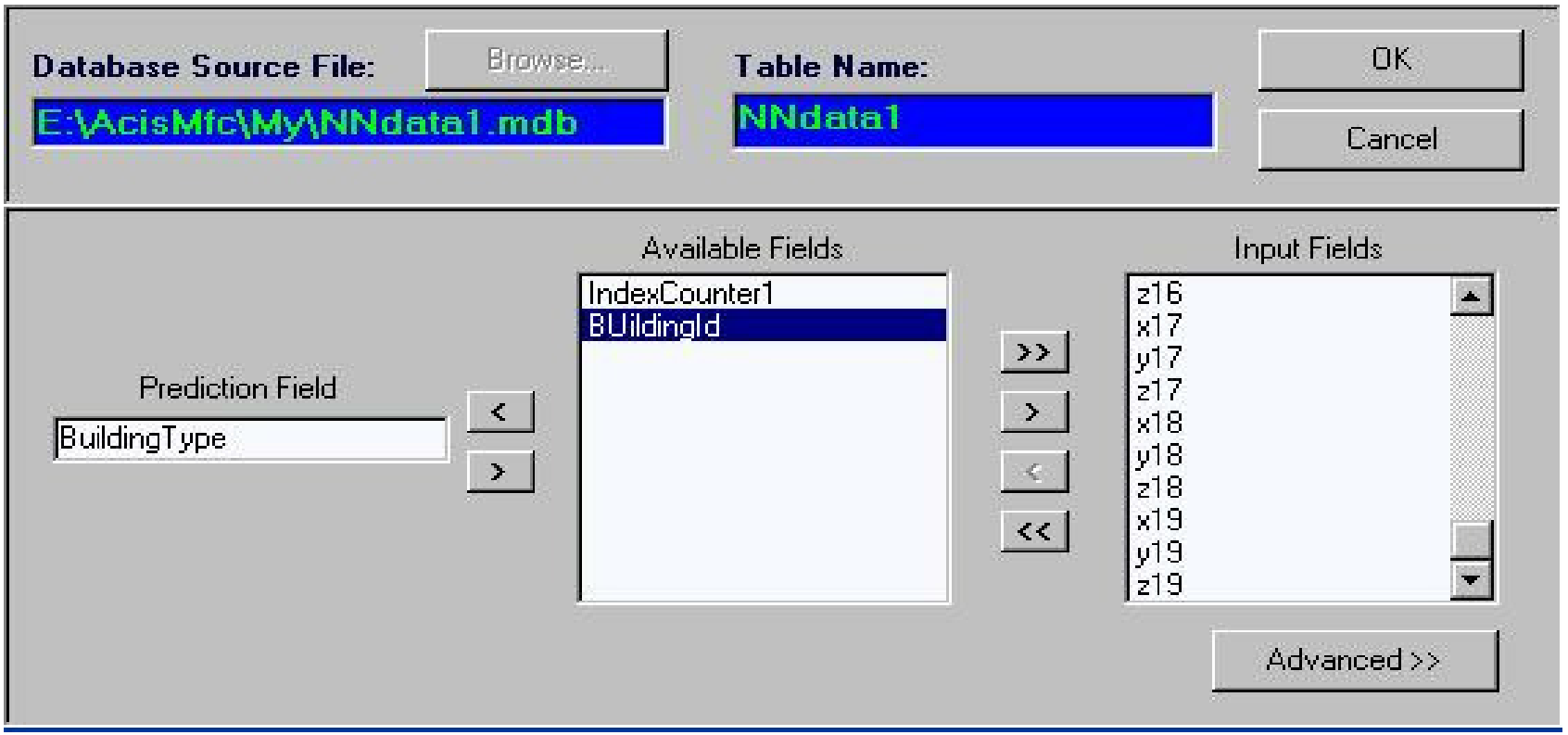
City Model

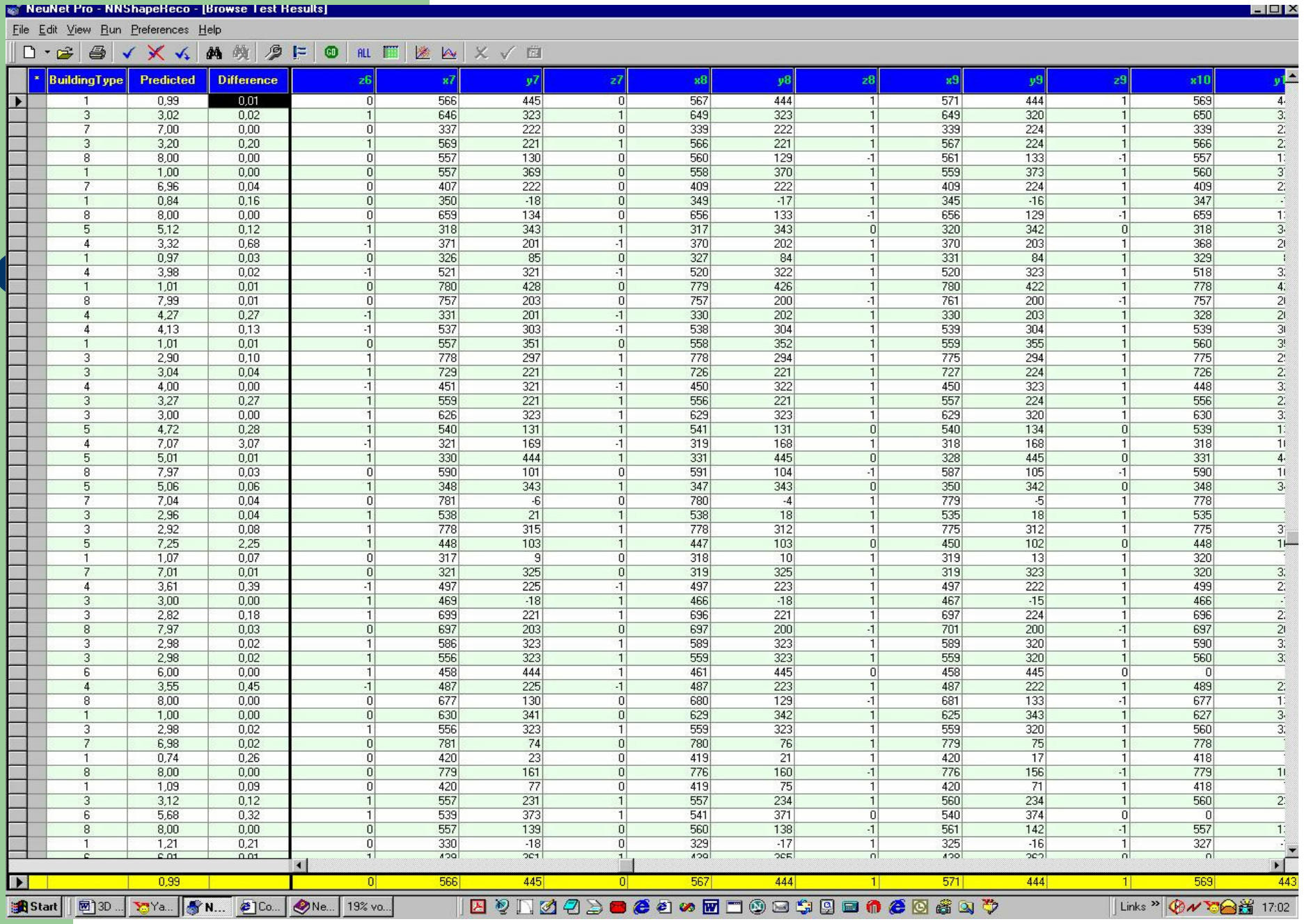


Neural Netweok (NN) approach

BUildingId,BuildingType,x0,y0,z0,x1,y1,z1,x2,y2,z2,x3,y3,z3,x4,y4,z4,x5,y5,z5,x6,y6,z6,x7,y7,z7,x8,y8,z8,x9,y9,z9,x10,y10,z10,x11,y11,z11,x12,y12,z12,x13,y13,z13,
x14,y14,z14635,1,606,477,-1,604,476,-1,603,478,-1,602,478,-1,602,481,-1,605,482,
1,606,477,0,605,482,0,604,480,1,605,476,1,603,478,0,604,476,0,602,478,0,602,479,1,602,481,0,0,0,00,0,00,0,00,0,00,0,0
307, 5,264, 464, -1,266,462,-1,263,460,-1,262,462,-
1,263,461,1,263,461,1,265,462,1,265,463,1,264,464,0,262,462,0,266,462,0,263,460,0,0,0,00,0,00,0,00,0,00,0,00,0,00,0,0
407,7,696,175,-1,696,173,-1,695,173,-1,695,172,-1,693,172,-1,693,175,-
1,696,173,0,696,175,0,694,175,1,694,173,1,695,172,0,695,173,0,694,172,1,693,175,0,693,172,0,0,0,00,0,00,0,00,0,00,0,0
538,3,579,310,-1,582,315,-1,584,313,-1,583,311,-1,586,310,-1,584,307,-
1,584,309,1,584,309,1,582,311,1,583,314,1,583,314,1,581,311,1,579,310,0,584,307,0,582,315,0,583,311,0,584,313,0,586,310,0,0,0,00,0,0
134,5,446,295,-1,446,293,-1,443,293,-1,443,295,-
1,443,294,1,443,293,1,445,293,1,445,294,1,446,295,0,443,295,0,446,293,0,443,293,0,0,0,00,0,00,0,00,0,00,0,00,0,00,0,0
601,1,516,535,-1,516,533,-1,514,533,-1,514,532,-1,511,532,-1,511,535,-
1,516,535,0,511,535,0,512,534,1,516,534,1,514,533,0,516,533,0,514,532,0,512,532,1,511,532,0,0,0,00,0,00,0,00,0,00,0,0
290,5,262,433,-1,264,435,-1,266,432,-1,264,431,-
1,264,432,1,265,432,1,263,434,1,263,434,1,262,433,0,264,431,0,264,435,0,266,432,0,0,0,00,0,00,0,00,0,00,0,00,0,00,0,0
0,1,266,175,-1,266,173,-1,264,173,-1,264,172,-1,261,172,-1,261,175,-
1,266,175,0,261,175,0,262,174,1,266,174,1,264,173,0,266,173,0,264,172,0,262,172,1,261,172,0,0,0,00,0,00,0,00,0,00,0,0
1,380,153,1,380,153,1,382,155,1,385,153,1,385,154,1,382,156,1,381,158,0,378,153,0,386,155,0,382,154,0,384,153,0,381,151,0,0,0,00,0,0
255,4,436,312,-1,433,310,-1,433,310,-1,432,310,-1,431,311,-1,431,312,-1,431,312,-1,433,315,-
1,433,313,1,432,312,1,433,311,1,434,312,1,436,312,0,433,315,0,433,310,0,431,312,0,433,310,0,431,312,0,431,311,0,432,310,0
679,1,726,441,-1,724,440,-1,723,442,-1,722,442,-1,722,445,-1,725,446,-
1,726,441,0,725,446,0,724,444,1,725,440,1,723,442,0,724,440,0,722,442,0,722,443,1,722,445,0,0,0,00,0,00,0,00,0,00,0,0
609,1,596,535,-1,596,533,-1,594,533,-1,594,532,-1,591,532,-
1,591,535,1,596,535,0,591,535,0,592,534,1,596,534,1,594,533,0,596,533,0,594,532,0,592,532,1,591,532,0,0,0,00,0,00,0,00,0,00,0,0
600,1,506,535,-1,506,533,-1,504,533,-1,504,532,-1,501,532,-1,501,535,-
1,506,535,0,501,535,0,502,534,1,506,534,1,504,533,0,506,533,0,504,532,0,502,532,1,501,532,0,0,0,00,0,00,0,00,0,00,0,0
411,7,633,74,-1,634,75,-1,635,74,-1,635,74,-1,636,72,-1,633,71,-
1,634,75,0,633,74,0,633,72,1,635,73,1,635,74,0,635,74,0,635,73,1,633,71,0,636,72,0,0,0,00,0,00,0,00,0,00,0,0
276,4,481,350,-1,483,353,-1,483,352,-1,484,352,-1,485,351,-1,485,350,-1,486,350,-1,483,348,-
1,483,349,1,484,350,1,483,351,1,482,350,1,481,350,0,483,348,0,483,353,0,485,350,0,483,352,0,486,350,0,485,351,0,484,352,0
109,4,263,219,-1,261,221,-1,261,221,-1,261,222,-1,262,223,-1,263,223,-1,263,224,-1,266,221,-
1,264,221,1,263,222,1,262,221,1,263,220,1,263,219,0,266,221,0,261,221,0,263,223,0,261,221,0,263,224,0,262,223,0,261,222,0
536,3,559,310,-1,562,315,-1,564,313,-1,563,311,-1,566,310,-1,564,307,-
1,564,309,1,564,309,1,562,311,1,563,314,1,563,314,1,561,311,1,559,310,0,564,307,0,562,315,0,563,311,0,564,313,0,566,310,0,0,0,00,0,0

Input Data


Database Source File:
Browse...
E:\AcisMfc\My\NNdata1.mdb
Table Name:
NNdata1
OK
Cancel
Available Fields
IndexCounter1
BUildingId
Input Fields
z16
x17
y17
z17
x18
y18
z18
x19
y19
z19
Prediction Field
BuildingType
<
>
>>
>
<
<<
Advanced >>



*	BuildingType	Predicted	Difference	z6	x7	y7	z7	x8	y8	z8	x9	y9	z9	x10	y1
	1	0,99	0,01	0	566	445	0	567	444	1	571	444	1	569	4
	3	3,02	0,02	1	646	323	1	649	323	1	649	320	1	650	3
	7	7,00	0,00	0	337	222	0	339	222	1	339	224	1	339	2
	3	3,20	0,20	1	569	221	1	566	221	1	567	224	1	566	2
	8	8,00	0,00	0	557	130	0	560	129	-1	561	133	-1	557	1
	1	1,00	0,00	0	557	369	0	558	370	1	559	373	1	560	3
	7	6,96	0,04	0	407	222	0	409	222	1	409	224	1	409	2
	1	0,84	0,16	0	350	-18	0	349	-17	1	345	-16	1	347	-
	8	8,00	0,00	0	659	134	0	656	133	-1	656	129	-1	659	1
	5	5,12	0,12	1	318	343	1	317	343	0	320	342	0	318	3
	4	3,32	0,68	-1	371	201	-1	370	202	1	370	203	1	368	2
	1	0,97	0,03	0	326	85	0	327	84	1	331	84	1	329	
	4	3,98	0,02	-1	521	321	-1	520	322	1	520	323	1	518	3
	1	1,01	0,01	0	780	428	0	779	426	1	780	422	1	778	4
	8	7,99	0,01	0	757	203	0	757	200	-1	761	200	-1	757	2
	4	4,27	0,27	-1	331	201	-1	330	202	1	330	203	1	328	2
	4	4,13	0,13	-1	537	303	-1	538	304	1	539	304	1	539	3
	1	1,01	0,01	0	557	351	0	558	352	1	559	355	1	560	3
	3	2,90	0,10	1	778	297	1	778	294	1	775	294	1	775	2
	3	3,04	0,04	1	729	221	1	726	221	1	727	224	1	726	2
	4	4,00	0,00	-1	451	321	-1	450	322	1	450	323	1	448	3
	3	3,27	0,27	1	559	221	1	556	221	1	557	224	1	556	2
	3	3,00	0,00	1	626	323	1	629	323	1	629	320	1	630	3
	5	4,72	0,28	1	540	131	1	541	131	0	540	134	0	539	1
	4	7,07	3,07	-1	321	169	-1	319	168	1	318	168	1	318	1
	5	5,01	0,01	1	330	444	1	331	445	0	328	445	0	331	4
	8	7,97	0,03	0	590	101	0	591	104	-1	587	105	-1	590	1
	5	5,06	0,06	1	348	343	1	347	343	0	350	342	0	348	3
	7	7,04	0,04	0	781	-6	0	780	-4	1	779	-5	1	778	
	3	2,96	0,04	1	538	21	1	538	18	1	535	18	1	535	
	3	2,92	0,08	1	778	315	1	778	312	1	775	312	1	775	3
	5	7,25	2,25	1	448	103	1	447	103	0	450	102	0	448	1
	1	1,07	0,07	0	317	9	0	318	10	1	319	13	1	320	
	7	7,01	0,01	0	321	325	0	319	325	1	319	323	1	320	3
	4	3,61	0,39	-1	497	225	-1	497	223	1	497	222	1	499	2
	3	3,00	0,00	1	469	-18	1	466	-18	1	467	-15	1	466	-
	3	2,82	0,18	1	699	221	1	696	221	1	697	224	1	696	2
	8	7,97	0,03	0	697	203	0	697	200	-1	701	200	-1	697	2
	3	2,98	0,02	1	586	323	1	589	323	1	589	320	1	590	3
	3	2,98	0,02	1	556	323	1	559	323	1	559	320	1	560	3
	6	6,00	0,00	1	458	444	1	461	445	0	458	445	0	0	
	4	3,55	0,45	-1	487	225	-1	487	223	1	487	222	1	489	2
	8	8,00	0,00	0	677	130	0	680	129	-1	681	133	-1	677	1
	1	1,00	0,00	0	630	341	0	629	342	1	625	343	1	627	3
	3	2,98	0,02	1	556	323	1	559	323	1	559	320	1	560	3
	7	6,98	0,02	0	781	74	0	780	76	1	779	75	1	778	
	1	0,74	0,26	0	420	23	0	419	21	1	420	17	1	418	
	8	8,00	0,00	0	779	161	0	776	160	-1	776	156	-1	779	1
	1	1,09	0,09	0	420	77	0	419	75	1	420	71	1	418	
	3	3,12	0,12	1	557	231	1	557	234	1	560	234	1	560	2
	6	5,68	0,32	1	539	373	1	541	371	0	540	374	0	0	
	8	8,00	0,00	0	557	139	0	560	138	-1	561	142	-1	557	1
	1	1,21	0,21	0	330	-18	0	329	-17	1	325	-16	1	327	-
	6	6,01	0,01	1	439	261	1	439	265	0	439	263	0	0	
		0,99		0	566	445	0	567	444	1	571	444	1	569	443

Spatial relationship

Spatial relationships among individuals and neighborhood objects is based upon followings:

i.	Proximity	Closest
ii.	Containment	Contained
iii.	Connectivity	Connect

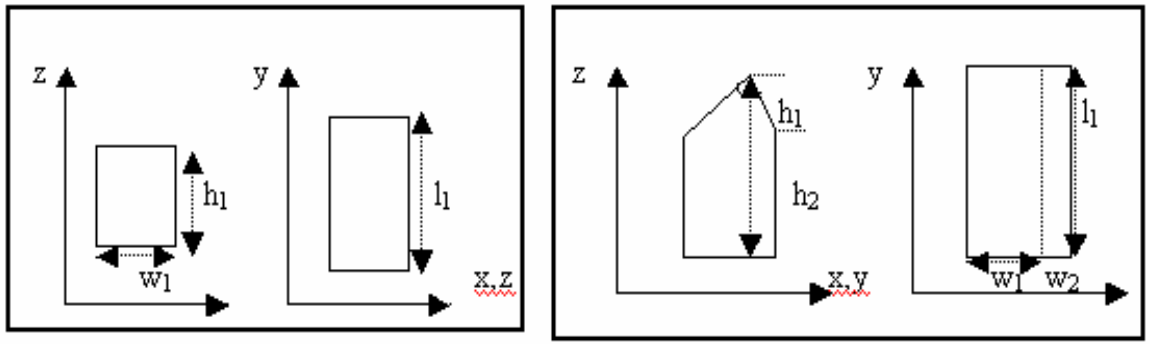
And can be divided into three levels:

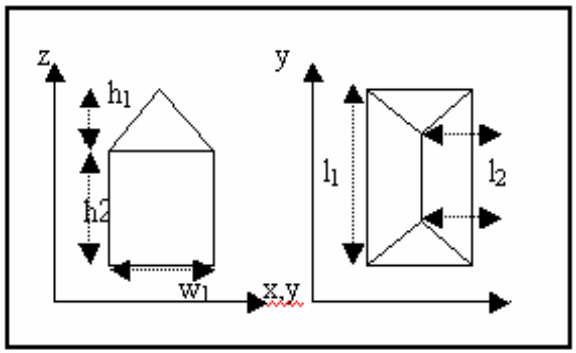
Micro Level

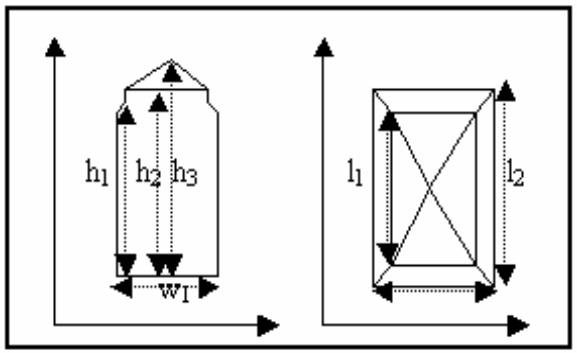
It applies to individual 3D which building can be characterized by:

- A set of measurable parameters is require to characterize a building, Number of minimum parameters require varies from building to buildings. Following are the details of parameters for few types of buildings.

Positional parameters





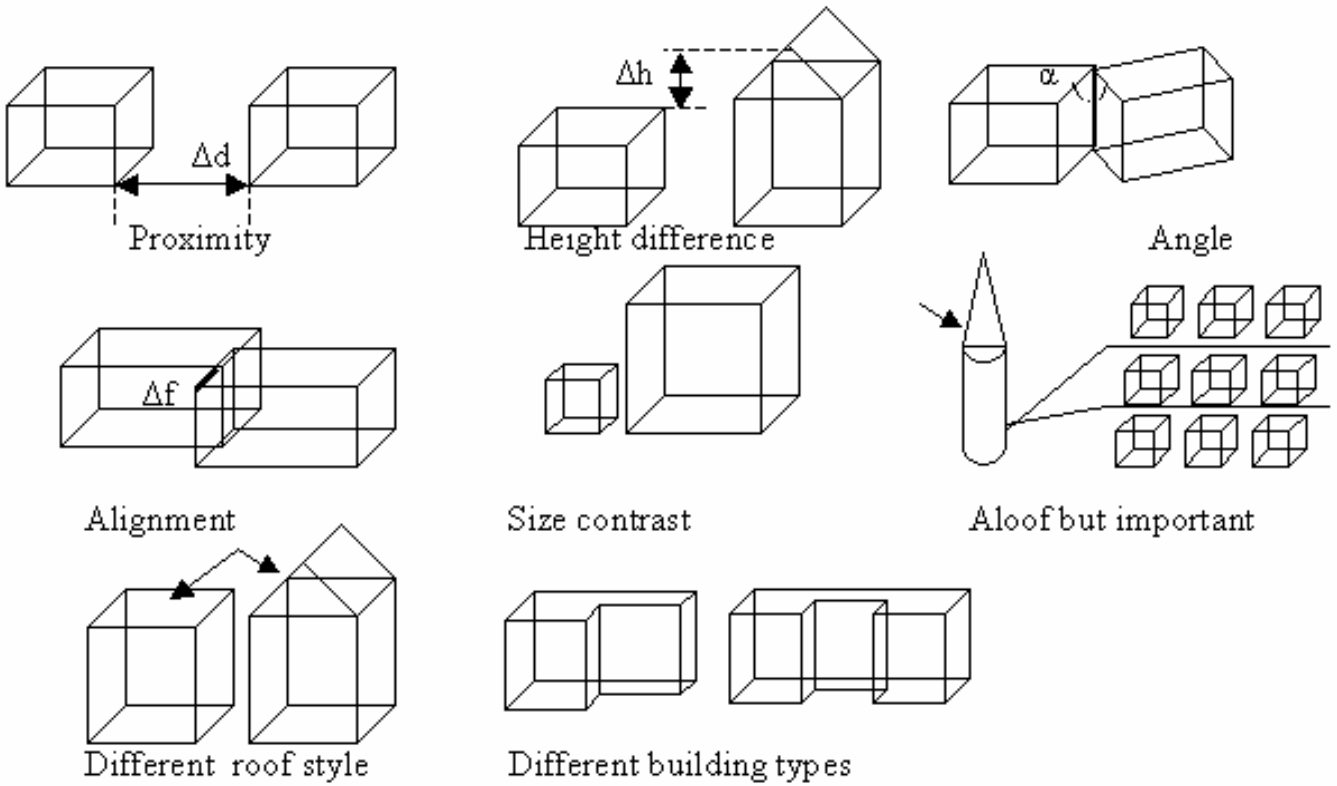


Other parameters

-
- Symmetry
- Regularity
- Orthogonal walls (in most cases)
- Roof types
- General Shapes
- L & U junctions
- parallel lines
- parallelism of faces

Meso level:

- It applies to an object in relation to its neighbors. Out of the three relations described above, proximity is very important relations among 3D buildings in neighborhood.
- Proximity can me obtained by measuring their mutual orientation, angle, distance between them and their height difference


Δd
Proximity
Δh
Height difference
α
Angle
Δf
Alignment
Size contrast
Aloof but important
Different roof style
Different building types

Macro level:

Applies to clusters of objects having similar properties such as settlement blocks and is based on visual grouping behaviors. These visual grouping behaviors are an important aspect in understanding images and maps and therefore has to be maintained. Macro level structure recognition helps to maintain this. Main emphasis is on

- Shape and size regularity.
- Height regularity.
- Regularity of roof structure and surface texture.
- Adjacency to other structures.
- Unique, deterministic features.
- Relative distributional density

Aggregation rules

Aggregation rules are based upon structure recogntion and can be divided into following groups as follow:

Linkage rules define the spatial relations between buildings that must exist for their aggregation.

- Proximity: Two buildings must be disjoint but within a certain distance of each other for their aggregation;
- -Alignment: Two buildings should be aligned or their alignment should not differ much and the difference must be below the permissible limits.
- Angle: Two buildings should have minimum angle between them if they are to be aggregated;
- Height: Two buildings should have minimum height difference if they are to be aggregated. Roof Type: Two buildings should have similar type of roof, i.e., planar or gable if they are to be aggregated;
- Adjacency: Two buildings must have their adjacent faces close if they are to be aggregated

Semantic rules: define the semantic relationships that must exist for aggregation. Semantic rules include relationships such as

- Class: Two buildings must be of the same class
- Structural: a group of objects form a common geographic or perceptual structure.

Orientational rules define the historical or local importance of the buildings. Important buildings such as monument, theatre, tower etc. reflect the city identity.

- Buildings which are of great importance should have their identities preserved at maximum of its extent**.** Therefore two buildings having historical importance should not be aggregated.
- If only one of the buildings is important, even then aggregation should not be done. Instead the important building should be exaggerated. The same rule applies to other local important buildings like TV towers, theatres, schools etc.

Based upon the above structural recognition, an algorithm for aggregation was developed. It is basically a nested set of If..Else statements. For example:

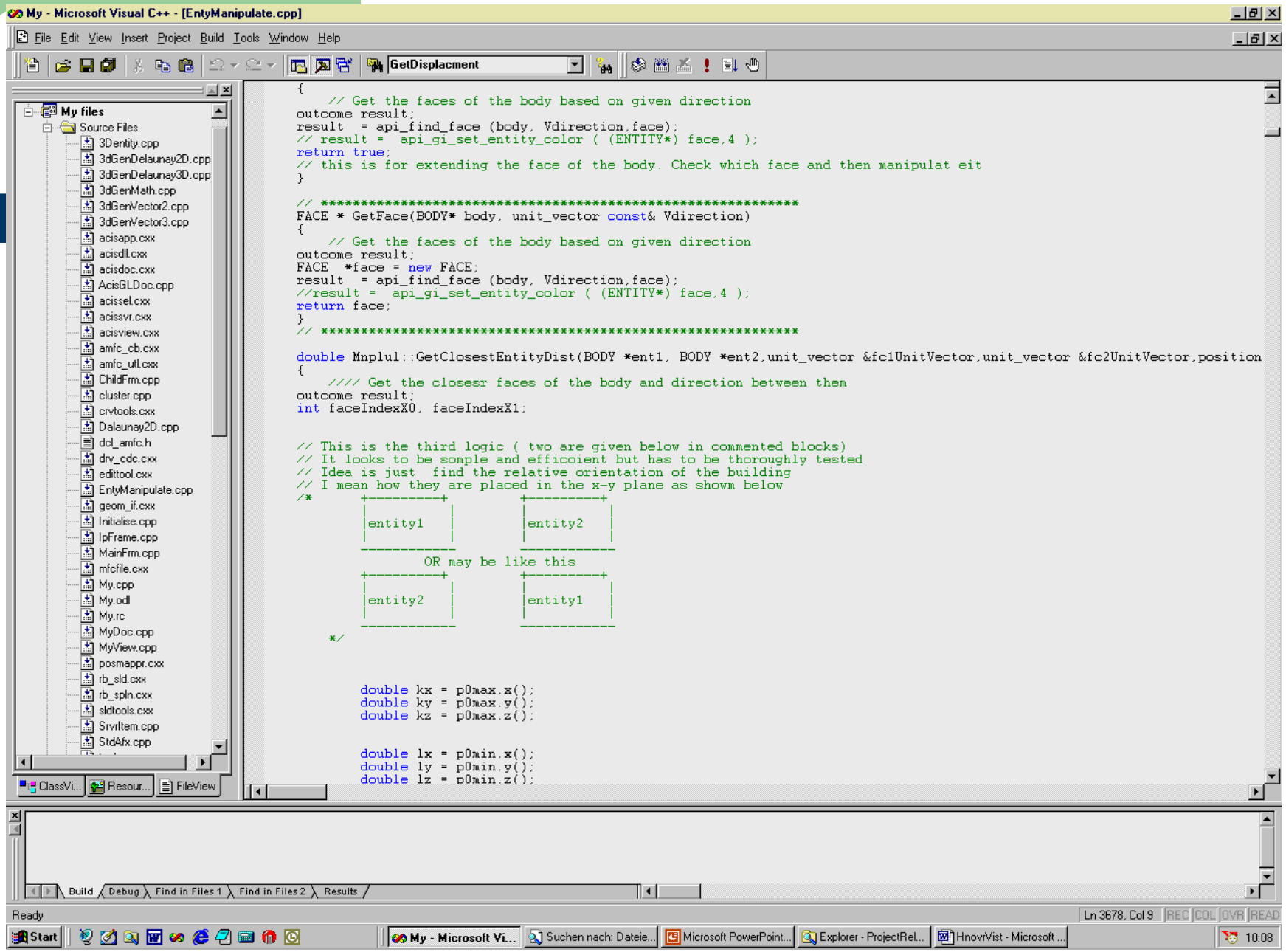
If $\boldsymbol{d}(o_i,o_j) < \Delta d_{\mathbf{min}}$ then AND

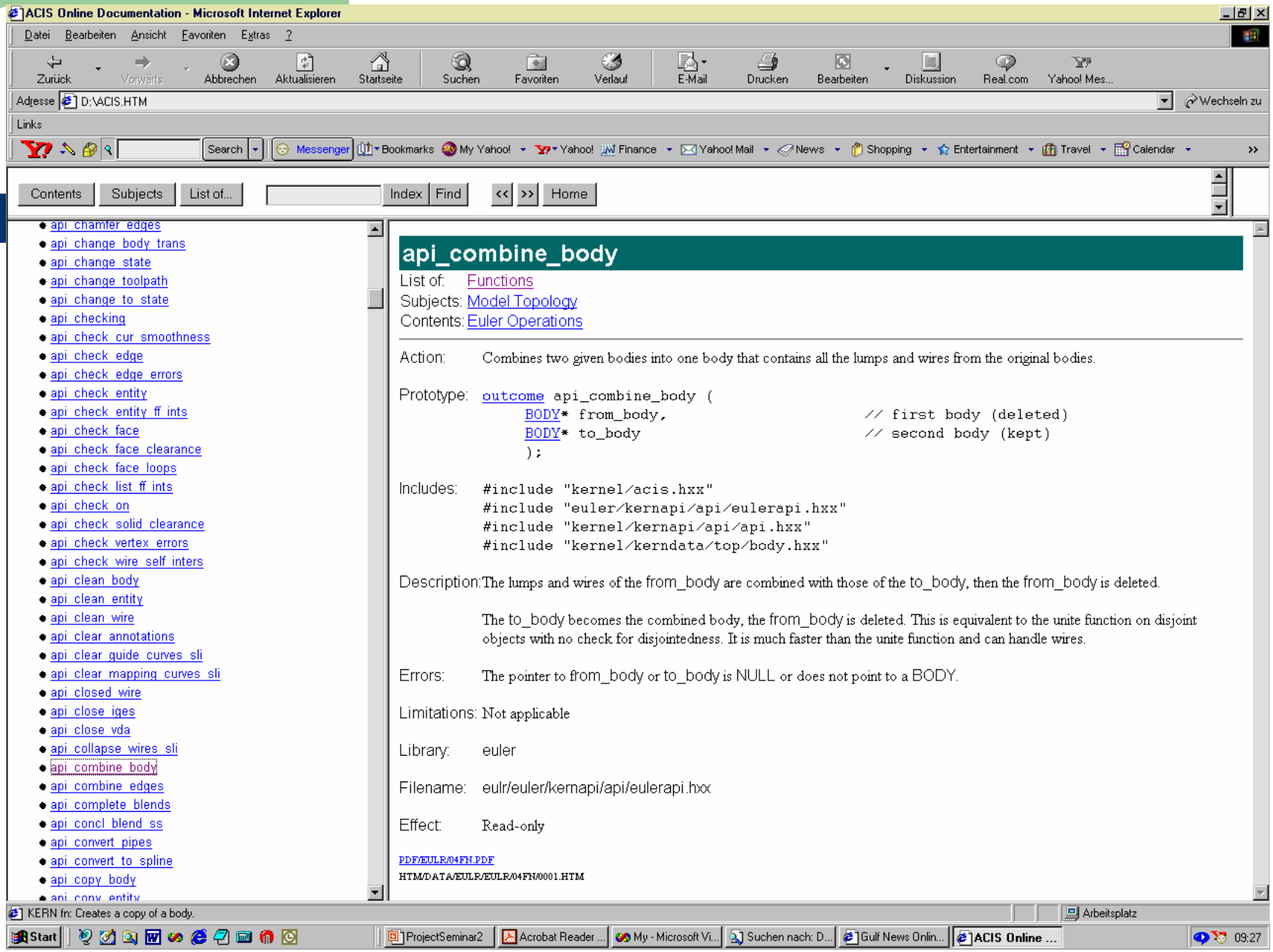
If $\boldsymbol{h}(o_i,o_j) < \Delta h_{\mathbf{min}}$ then AND

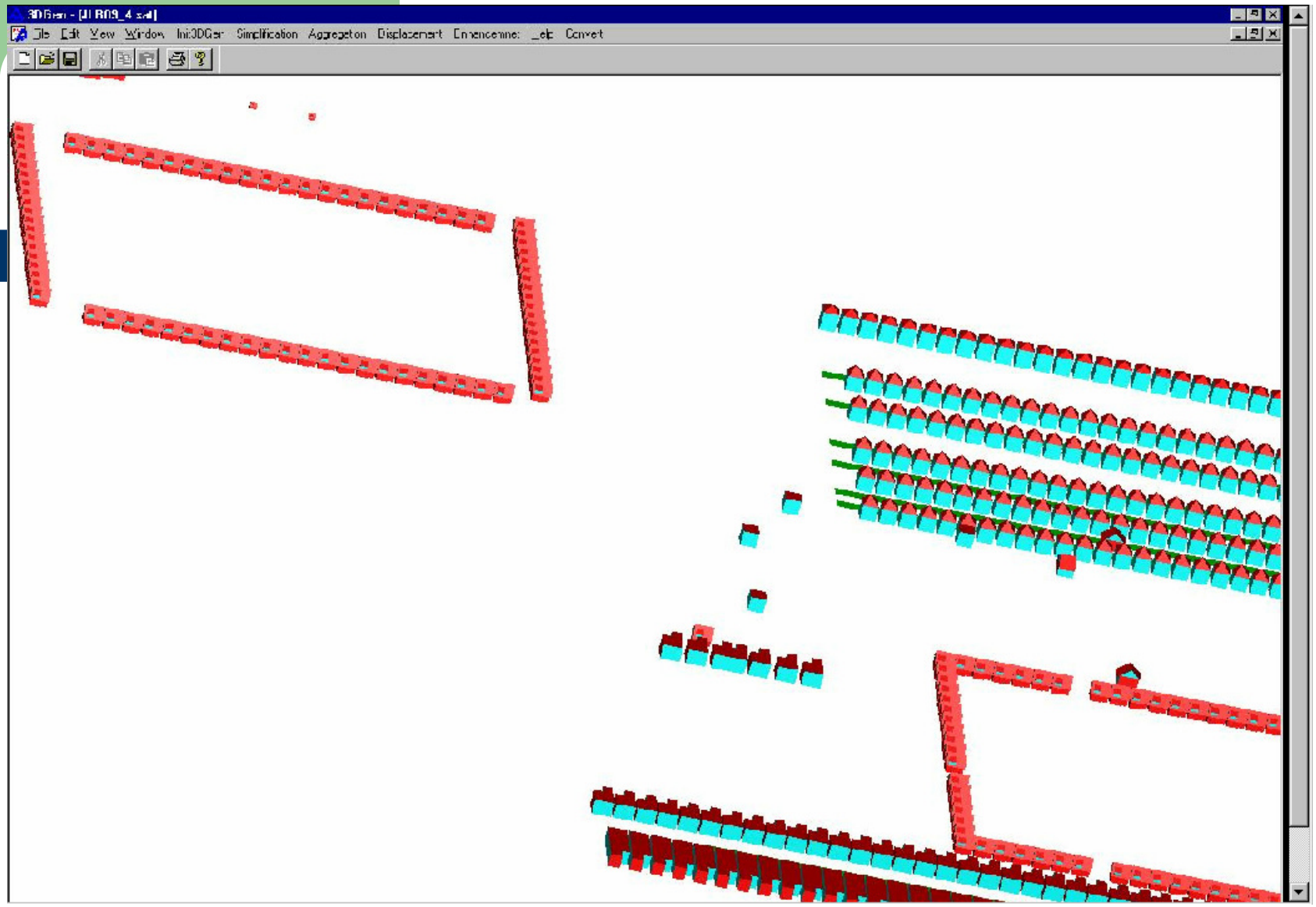
If $\boldsymbol{roof\_type_1} = \boldsymbol{roof\_type_2}$ then AND

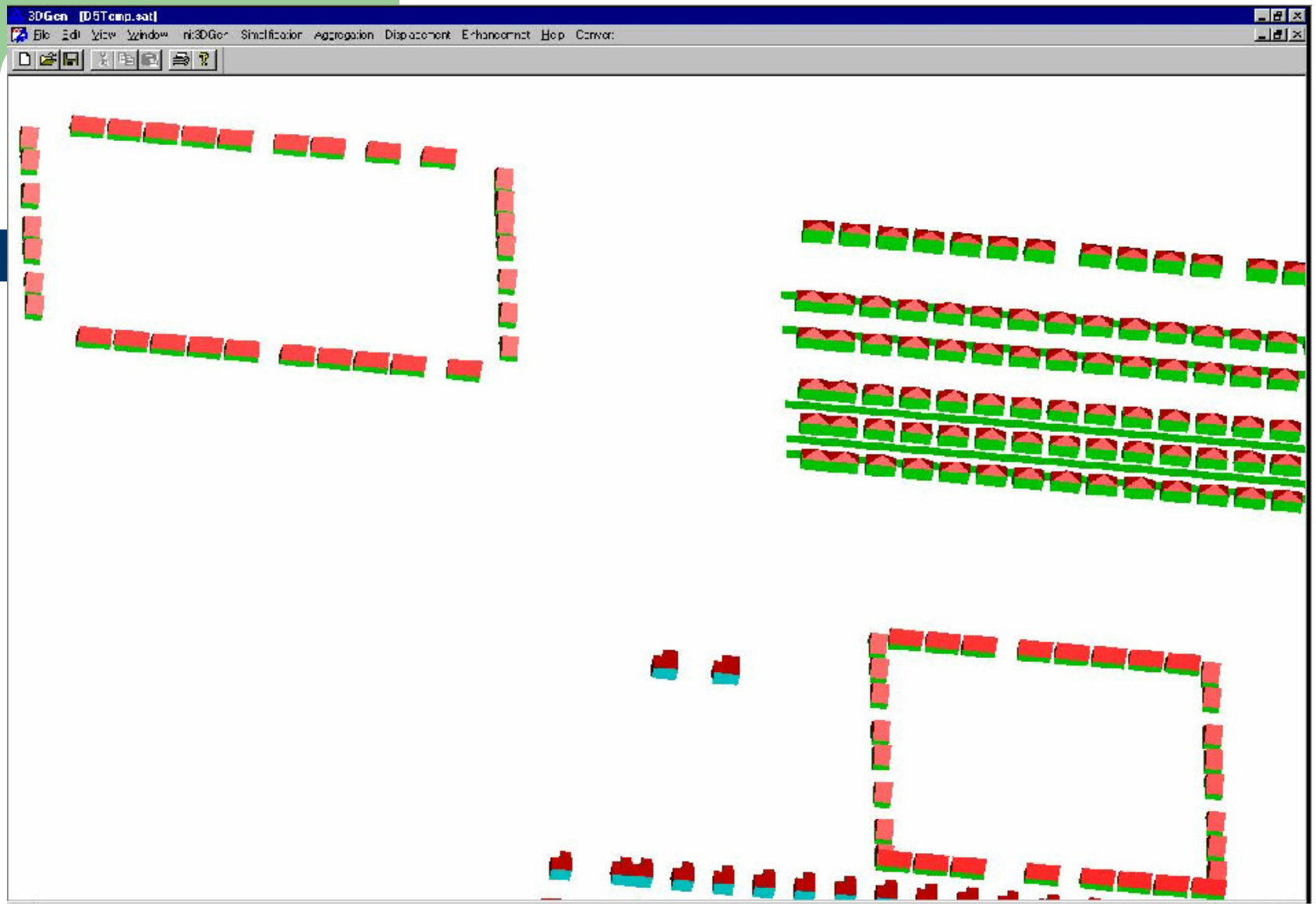
If $\Delta \mathbf{A}(o_i,o_j) < \mathbf{A_{min}}$ then **aggregate**

…


My - Microsoft Visual C++ - [EntyManipulate.cpp]
File Edit View Insert Project Build Tools Window Help
GetDisplacment
My files
Source Files
3Dentity.cpp
3dGenDelaunay2D.cpp
3dGenDelaunay3D.cpp
3dGenMath.cpp
3dGenVector2.cpp
3dGenVector3.cpp
acisapp.cxx
acisdll.cxx
acisdoc.cxx
AcisGLDoc.cpp
acissel.cxx
acissvr.cxx
acisview.cxx
amfc_cb.cxx
amfc_utl.cxx
ChildFrm.cpp
cluster.cpp
crvtools.cxx
Dalaunay2D.cpp
dcl_amfc.h
drv_cdc.cxx
edittool.cxx
EntyManipulate.cpp
geom_if.cxx
Initialise.cpp
IpFrame.cpp
MainFrm.cpp
mfcfile.cxx
My.cpp
My.odl
My.rc
MyDoc.cpp
MyView.cpp
posmappr.cxx
rb_sld.cxx
rb_spln.cxx
sldtools.cxx
SrvrItem.cpp
StdAfx.cpp
ClassVi...
Resour...
FileView
{
// Get the faces of the body based on given direction
outcome result;
result = api_find_face (body, Vdirection,face);
// result = api_gi_set_entity_color ((ENTITY*) face,4);
return true;
// this is for extending the face of the body. Check which face and then manipulat eit
}
// **
FACE * GetFace(BODY* body, unit_vector const& Vdirection)
{
// Get the faces of the body based on given direction
outcome result;
FACE *face = new FACE;
result = api_find_face (body, Vdirection,face);
//result = api_gi_set_entity_color ((ENTITY*) face,4);
return face;
}
// **
double Mnplul::GetClosestEntityDist(BODY *ent1, BODY *ent2,unit_vector &fc1UnitVector,unit_vector &fc2UnitVector,position
{
//// Get the closesr faces of the body and direction between them
outcome result;
int faceIndexX0, faceIndexX1;
// This is the third logic (two are given below in commented blocks)
// It looks to be somple and efficoient but has to be thoroughly tested
// Idea is just find the relative orientation of the building
// I mean how they are placed in the x-y plane as shown below
/*
+---------+ +---------+
entity1		entity2
---------- ----------
OR may be like this
+---------+ +---------+
entity2		entity1
---------- ----------
*/
double kx = p0max.x();
double ky = p0max.y();
double kz = p0max.z();
double lx = p0min.x();
double ly = p0min.y();
double lz = p0min.z();
Build Debug Find in Files 1 Find in Files 2 Results
Ready
Ln 3678, Col 9
REC COL OVR READ
Start
My - Microsoft Vi...
Suchen nach: Dateie...
Microsoft PowerPoint...
Explorer - ProjectRel...
HnovrVist - Microsoft ...
10:08


ACIS Online Documentation - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Vorwärts Abbrechen Aktualisieren Startseite Suchen Favoriten Verlauf E-Mail Drucken Bearbeiten Diskussion Real.com Yahoo! Mes...
Adresse D:\ACIS.HTM
Wechseln zu
Links
Search Messenger Bookmarks My Yahoo! Yahoo! Finance Yahoo! Mail News Shopping Entertainment Travel Calendar
Contents Subjects List of... Index Find << >> Home
api_chamfer_edges
api_change_body_trans
api_change_state
api_change_toolpath
api_change_to_state
api_checking
api_check_cur_smoothness
api_check_edge
api_check_edge_errors
api_check_entity
api_check_entity_ff_ints
api_check_face
api_check_face_clearance
api_check_face_loops
api_check_list_ff_ints
api_check_on
api_check_solid_clearance
api_check_vertex_errors
api_check_wire_self_inters
api_clean_body
api_clean_entity
api_clean_wire
api_clear_annotations
api_clear_guide_curves_sli
api_clear_mapping_curves_sli
api_closed_wire
api_close_iges
api_close_vda
api_collapse_wires_sli
api_combine_body
api_combine_edges
api_complete_blends
api_concl_blend_ss
api_convert_pipes
api_convert_to_spline
api_copy_body
api_copy_entity
api_combine_body
List of: Functions
Subjects: Model Topology
Contents: Euler Operations
Action: Combines two given bodies into one body that contains all the lumps and wires from the original bodies.
Prototype: outcome api_combine_body (
BODY* from_body, // first body (deleted)
BODY* to_body // second body (kept)
);
Includes: #include "kernel/acis.hxx"
#include "euler/kernapi/api/eulerapi.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "kernel/kerndata/top/body.hxx"
Description: The lumps and wires of the from_body are combined with those of the to_body, then the from_body is deleted.
The to_body becomes the combined body, the from_body is deleted. This is equivalent to the unite function on disjoint objects with no check for disjointedness. It is much faster than the unite function and can handle wires.
Errors: The pointer to from_body or to_body is NULL or does not point to a BODY.
Limitations: Not applicable
Library: euler
Filename: eulr/euler/kernapi/api/eulerapi.hxx
Effect: Read-only
PDF/EULR/04FN.PDF
HTM/DATA/EULR/EULR/04FN/0001.HTM
KERN fn: Creates a copy of a body.
Arbeitsplatz
Start
ProjectSeminar2
Acrobat Reader ...
My - Microsoft Vi...
Suchen nach: D...
Gulf News Onlin...
ACIS Online ...
09:27


File Edit View Window Ini3DGen Simplification Aggregation Displacement Environnement Help Convert


3DGen [D5Temp.sat]
File Edit View Window ni3DGen Simplification Aggregation Displacement Enhancement Help Convert

Future work:

Macro level spatial relation based upon visual grouping will be studied and consequently quantified to judge the quality of generalization. These relation are based upon:

- Parallelism
- Continuity
- Closure
- Proximity
- Orientation

Thanks!