Methods for photographic radiometry, modeling of light transport and material appearance

Radiosity & advanced global illumination

Theodore Tsesmelis | 2018-09-08 | 3DV 2018, Verona

Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

Practical examples

Applications

Conclusion & future work

Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

Practical examples

Applications

Conclusion & future work

Measuring light…

Measuring light…

Precalibrated Cameras

- Light-to-pixel function (look-up table to a known source)

- Every camera is different
- Light-to-pixel function
  - Camera properties
  - Camera-to-camera differences

- No as trivial as it looks
- Accuracy questionable

Measuring light…

Precalibrated Cameras

- ❑ Light-to-pixel function (look-up table to a known source)

- ❑ Every camera is different
- ❑ Light-to-pixel function
  - ❑ Camera properties
  - ❑ Camera-to-camera differences

- ❑ No as trivial as it looks
- ❑ Accuracy questionable

Luxmeters

- ❑ Point-to-point → sparse measurements
- ❑ Time consuming for large areas
- ❑ Manually operated
- ❑ Relative cost/accuracy

Measuring light…

Precalibrated Cameras

- ❑ Light-to-pixel function (look-up table to a known source)

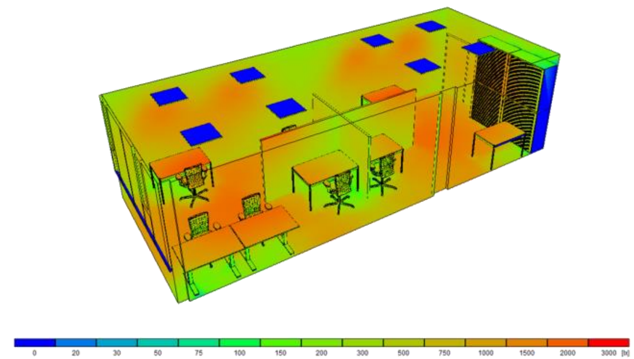
- ❑ Every camera is different
- ❑ Light-to-pixel function
    - ❑ Camera properties
    - ❑ Camera-to-camera differences

- ❑ No as trivial as it looks
- ❑ Accuracy questionable

Luxmeters

- ❑ Point-to-point → sparse measurements
- ❑ Time consuming for large areas
- ❑ Manually operated
- ❑ Relative cost/accuracy

Simulation software



- ❑ Rendering algorithms → simulate light propagation
- ❑ Offline / Time consuming
- ❑ Manual input, need of scene information in advance
    - ❑ Geometry (closed form), CAD models
    - ❑ Inaccuracies in the CAD model
    - ❑ Lights positioning and intensity, characteristics
    - ❑ Reflectance and material properties
- ❑ Relatively high accuracy and dense mapping

Rendering techniques

- Computer graphics
  - Simulate light
  - Realistic, lifelike renderings → games, movies, etc…
    - Radiosity
    - Ray tracing
    - Instant radiosity, VPLs
    - Photon mapping
    - Screen space
    - etc….
  - Actual light measurements
- Lighting design & modeling field
  - Relux, Dialux, AGi32 → radiosity
  - HILITE, LiteMaker[1]
    - photon mapping
    - academic prototypes
  - Manual input (geometry, lighting, material properties)

<u>*Blender 3D Architect*</u>

Source[1]

1. K. Krsl, C. Luksch, M. Schwrzler, and M. Wimmer. LiteMaker: Interactive Luminaire Development using Progressive Photon Tracing and Multi-Resolution Upsampling. Vision, Modeling & Visualization. The Eurographics Association, 2017.

Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

Practical examples

Applications

Conclusion & future work

Radiosity

- ❑ Simple, in principle
- ❑ Physically-based illumination algorithm
- ❑ Indirect illumination effects → global illumination
- ❑ Is computed in the *object-space* → view independent
- ❑ Assume, Lambertian surfaces

photograph

simulation

Pictures from: Light Measurement Laboratory Cornell University, Program for Computer Graphics

Radiosity

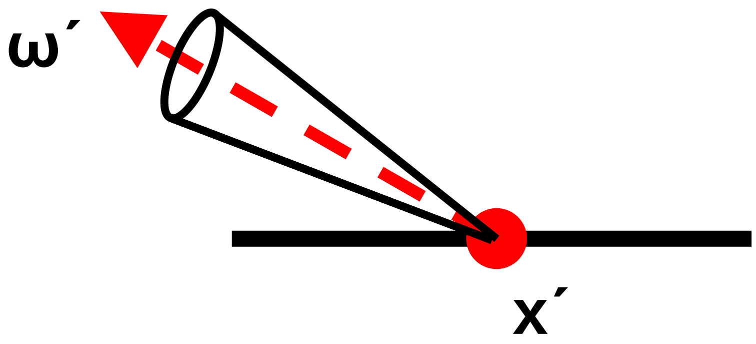
- ❑ Simple, in principle
- ❑ Physically-based illumination algorithm
- ❑ Indirect illumination effects → global illumination
- ❑ Is computed in the *object-space* → view independent
- ❑ Assume, Lambertian surfaces

photograph

simulation

$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega')L(x,\omega)G(x,x')V(x,x')\, dA$$

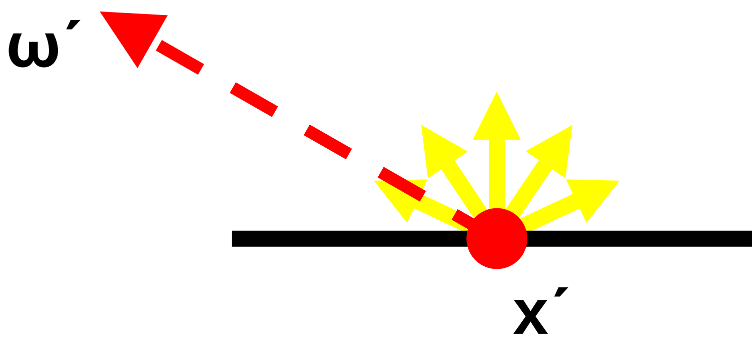
Rendering equation



$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

The radiance from a point on a surface in a given direction ω´

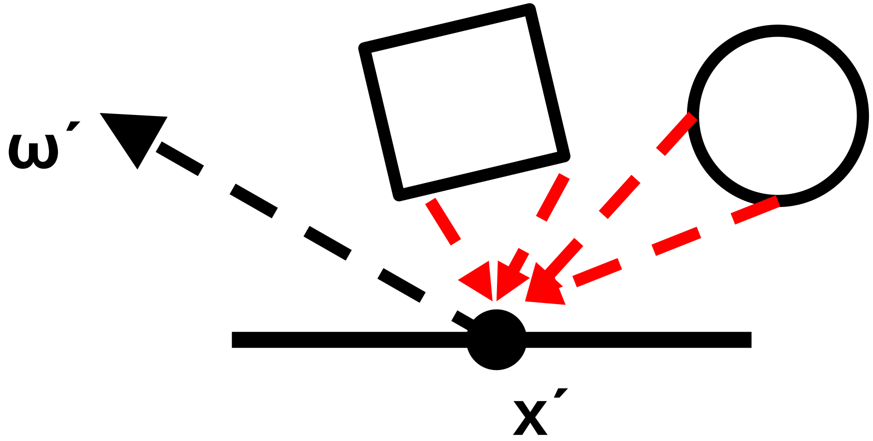
Rendering equation



$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

The emitted radiance from a point: E considered to be zero or non-zero depending whether it is a light source or not.

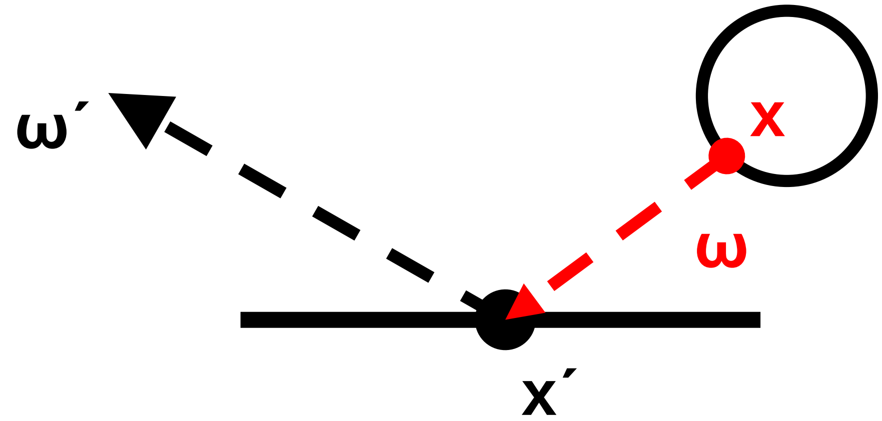
Rendering equation



$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

The contribution from all other surfaces of the scene.

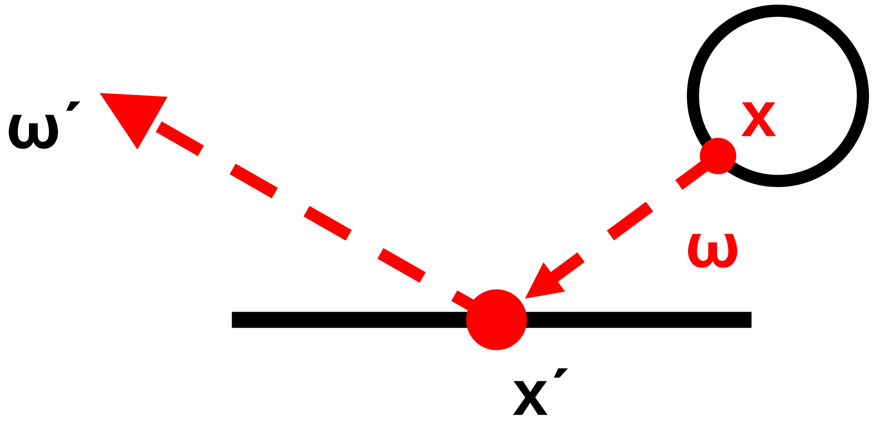
Rendering equation

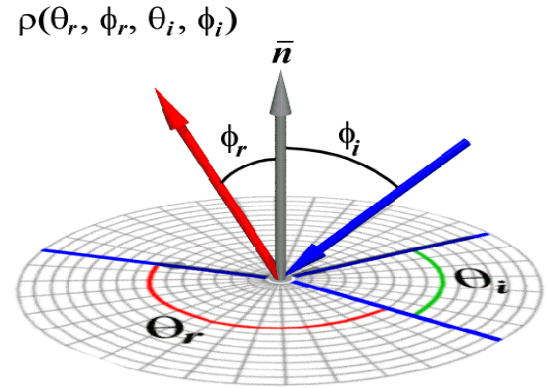


$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega')L(x,\omega)G(x,x')V(x,x')dA$$

For each other x point compute the radiance with direction ω (from x to x´).

Rendering equation

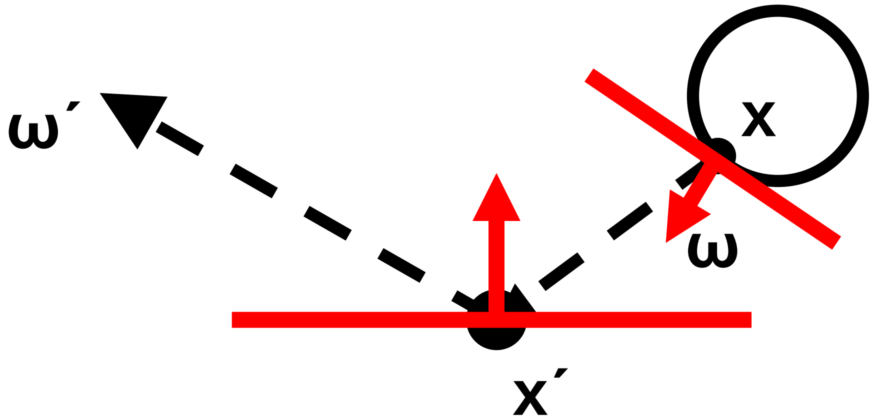




$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x') dA$$

Scale the contribution by the reflectivity (BRDF) at the surface x´.

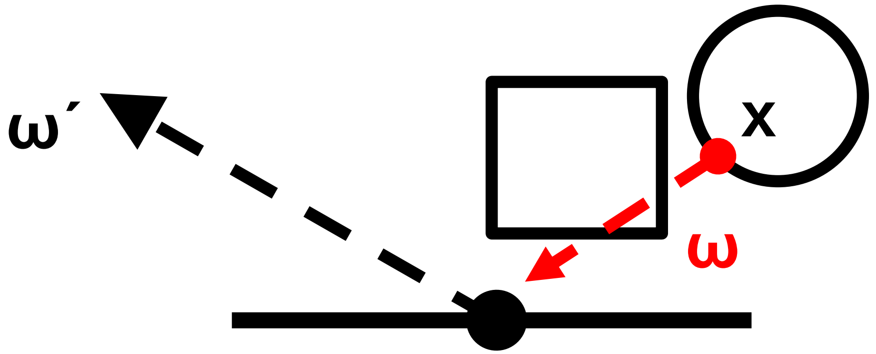
Rendering equation



$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega')L(x,\omega)G(x,x')V(x,x')dA$$

For each x compute the G(x,x´), which describes the geometric relationship between the two surfaces at x and x´.

Rendering equation



$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

For each x compute V(x,x´), the visibility between x and x´. 1 in case the two surfaces are visible to each other or 0 otherwise.

Radiosity equation

$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

Lambertian assumption

(perfectly diffuse surfaces, not directional)

$$B_{x'} = E_{x'} + \rho_{x'} \int B_x \; G(x, x') V(x, x')$$

Source[2]

[2] MIT EECS 6.837, Durand and Cutler

Radiosity equation

$$L(x',\omega') = E(x',\omega') + \int \rho_{x'}(\omega,\omega') L(x,\omega) G(x,x') V(x,x') dA$$

Lambertian assumption

(perfectly diffuse surfaces, not directional)

$$B_{x'} = E_{x'} + \rho_{x'} \int B_x \, G(x,x') V(x,x')$$

form factors

Radiosity equation

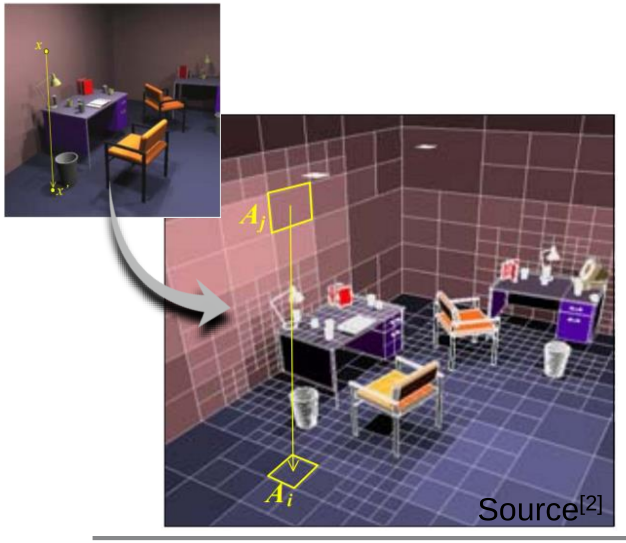
$$L(x', \omega') = E(x', \omega') + \int \rho_{x'}(\omega, \omega') L(x, \omega) G(x, x') V(x, x') dA$$

Lambertian assumption

(perfectly diffuse surfaces, not directional)

$$B_{x'} = E_{x'} + \rho_{x'} \int B_x \, G(x, x') V(x, x')$$

form factors

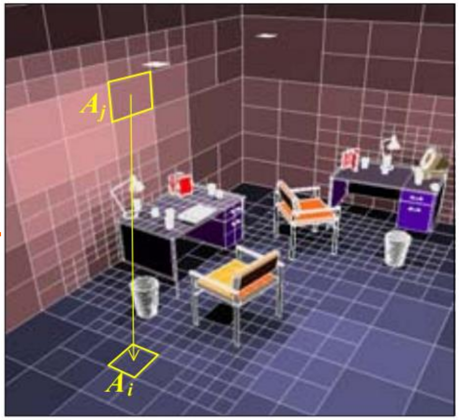


Source[2]

$$B_i = E_i + \rho_i \sum_{j=1}^{n} F_{ij} B_j$$

[2] MIT EECS 6.837, Durand and Cutler

Radiosity matrix

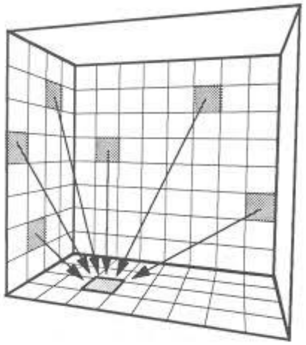


$$B_i = E_i + \rho_i \sum_{j=1}^{n} F_{ij} B_j$$

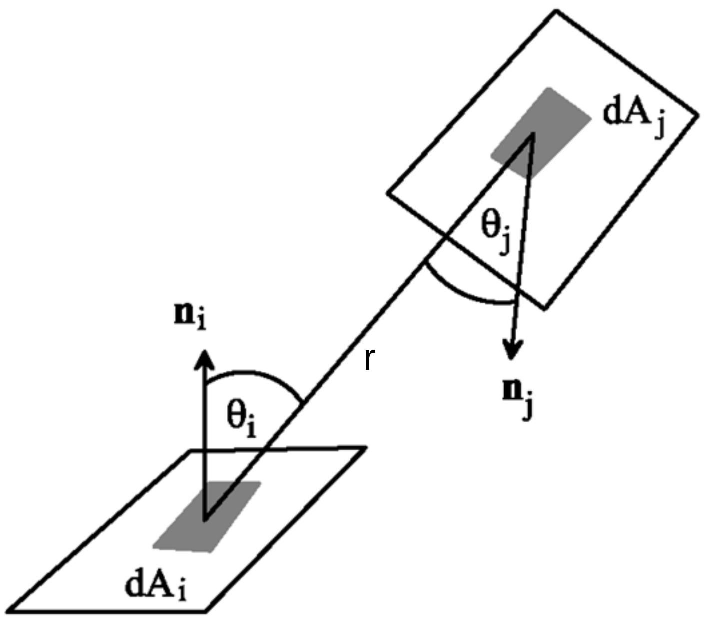
n simultaneous linear equations, with ***n*** unknown $\boldsymbol{B_j}$ radiosity values which can be written in a matrix form as:

$$\begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_i \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \\ \vdots \\ E_n \end{bmatrix} + \begin{bmatrix} & & & \\ \rho_i F_{i1} & \rho_i F_{i2} & \dots & \rho_i F_{in} \\ & & & \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ \vdots \\ B_n \end{bmatrix} \Rightarrow \begin{bmatrix} 1-\rho_1 F_{11} & -\rho_1 F_{12} & \cdots & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1-\rho_1 F_{22} & \cdots & -\rho_2 F_{2n} \\ \vdots & \vdots & \ddots & \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \cdots & 1-\rho_1 F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{bmatrix}$$

A solution yields a single radiosity value $\boldsymbol{B_j}$ for each patch in the environment by gathering radiosities from all other patches, a view-independent solution.



Form factors



The form factors are defined as the fraction of energy leaving one surface and reaches another surface. It is purely geometric relationship, independent of viewpoint or surface attributes.

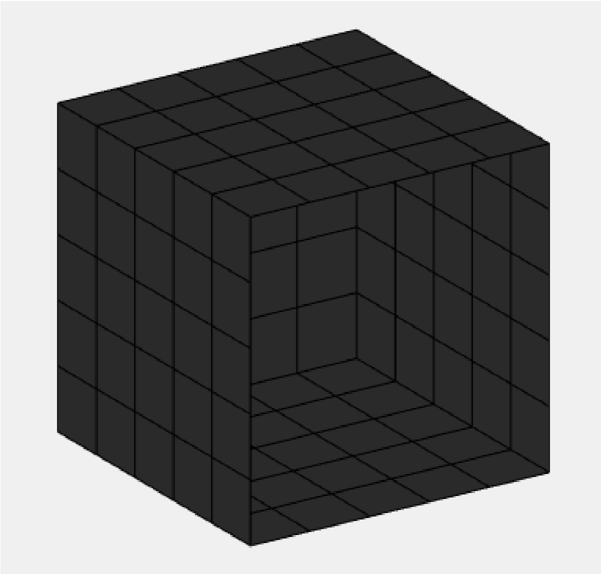
- **Geometry**
- **Visibility**

Analytical Solution:

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r^2} V_{ij} \, dA_j \, dA_i$$

Radiosity in practice…

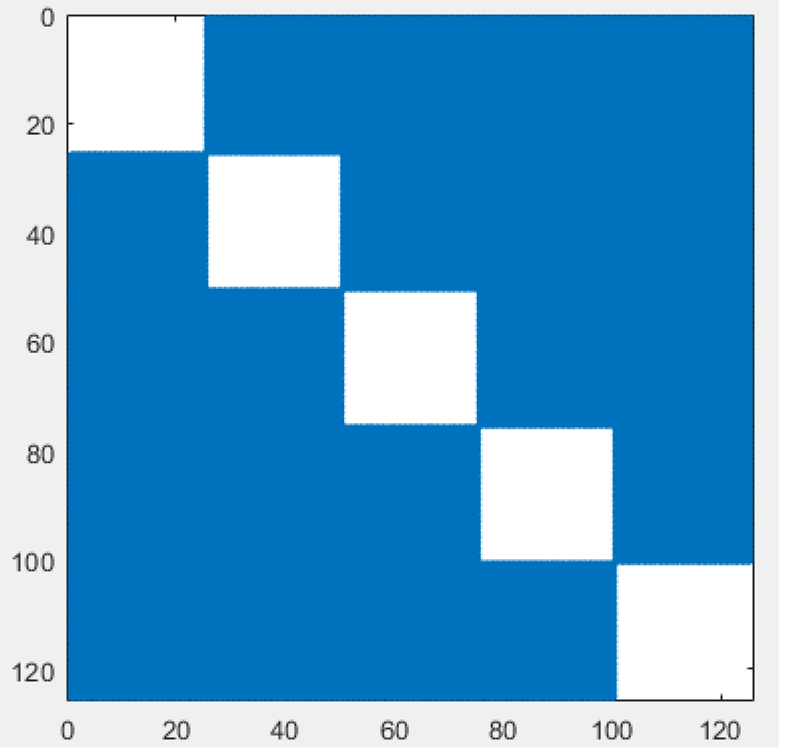
```
% Choose mosaic resolution
n = 5;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Construct centerpoints of the mosaic tiles.
% The letter d denotes the length of the side of a pixel.
d        = 2/n;
tmp      = -1-d/2 + (1:n)*d;

% Initialize centerpoint coordinate matrices
Xmat = zeros(n^2,5);
Ymat = zeros(n^2,5);
Zmat = zeros(n^2,5);

% Construct the centerpoints for all the tiles in all the five walls.
% The ordering of the five walls below fixes the indexing of all the tiles
% using just one number running from 1 to 5*(n^2).

% The back wall
[X,Z]      = meshgrid(tmp);
Xmat(:,1)  = X(:);
Zmat(:,1)  = Z(:);
Ymat(:,1)  = ones(n^2,1);

% Roof
[X,Y] = meshgrid(tmp);
Xmat(:,2)  = X(:);
Ymat(:,2)  = Y(:);
Zmat(:,2)  = ones(n^2,1);

% Floor
Xmat(:,3)  = X(:);
Ymat(:,3)  = Y(:);
Zmat(:,3)  = -ones(n^2,1);

...
...
...
...
```



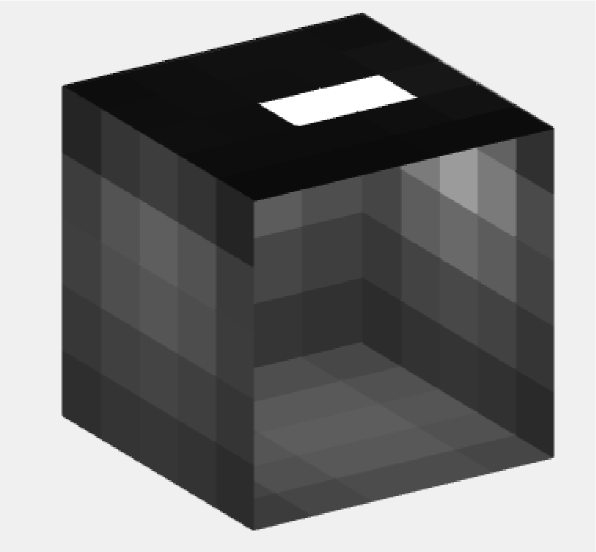
**Souce code initially written from Samuli Siltanen and modified by Theodore Tsesmelis*

Radiosity in practice…

```
...
...
...
% Initialize the matrix
F = zeros(5*n^2);

% From the roof (jjj) to the back wall (iii)
for iii = 1:n^2
    for jjj = 1:n^2
        % Centerpoint of the current pixel in the back wall
        piii = [Xmat(iii,1);Ymat(iii,1);Zmat(iii,1)];
        % Centerpoint of the current pixel in the roof
        pjjj = [Xmat(jjj,2);Ymat(jjj,2);Zmat(jjj,2)];
        % Distance between the points
        difvec = piii-pjjj;
        r      = norm(difvec);
        % View angles
        tmp2   = difvec/r;
        cosjjj = abs(tmp2(3));
        cosiii = abs(tmp2(2));
        % Calculate element of F
        F(iii,n^2+jjj) = cosiii*cosjjj/(pi*(r)^2);
    end
end
% From the floor (jjj) to the back wall (iii)
for iii = 1:n^2
    for jjj = 1:n^2
        % Centerpoint of the current pixel in the back wall
        piii = [Xmat(iii,1);Ymat(iii,1);Zmat(iii,1)];
        % Centerpoint of the current pixel in the roof
        pjjj = [Xmat(jjj,3);Ymat(jjj,3);Zmat(jjj,3)];
        % Distance between the points
        difvec = piii-pjjj;
        r      = norm(difvec);
        % View angles
        tmp2   = difvec/r;
        cosjjj = abs(tmp2(3));
        cosiii = abs(tmp2(2));
        % Calculate element of F
        F(iii,2*n^2+jjj) = cosiii*cosjjj/(pi*(r)^2);
    end
end
...
...
...
```

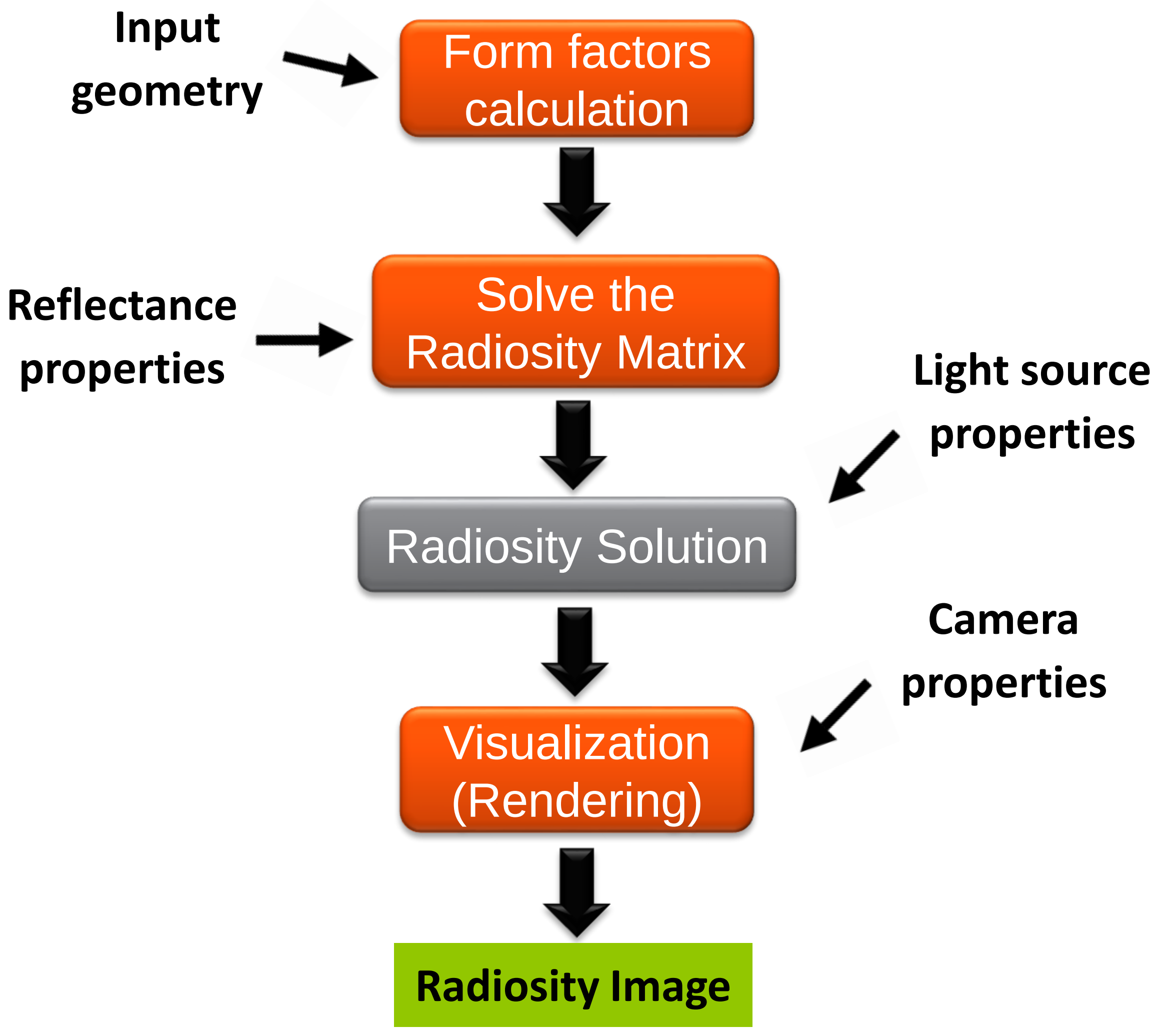


Radiosity in practice…



```
...
...
...
% Add the contribution of the area of each pixel
F = (d^2)*F;

% Use symmetry to finish the construction of F
F = F+F.';

% Construct the right hand side Evec of the radiosity equation. Evec
% describes the contribution of emitted light in the scene. For example,
% each pixel belonging to a lamp in the virtual space causes a positive
% element in Evec.
Evec   = zeros(5*n^2,1);
indvec = repmat(logical(0),size(Evec));
indvec(n^2+[1:n^2]) = sqrt((Xmat(:,2)-.3).^2+Ymat(:,2).^2)<.3;
Evec(indvec) = 1;

% Solve for color vector.
% The parameter rho adjusts the surface material (how much incoming light
% is reflected away from a patch, 0<rho<=1)
rho = 1;
Bvec = inv((eye(5*n^2)-rho*F))*Evec;
```

Radiosity overview



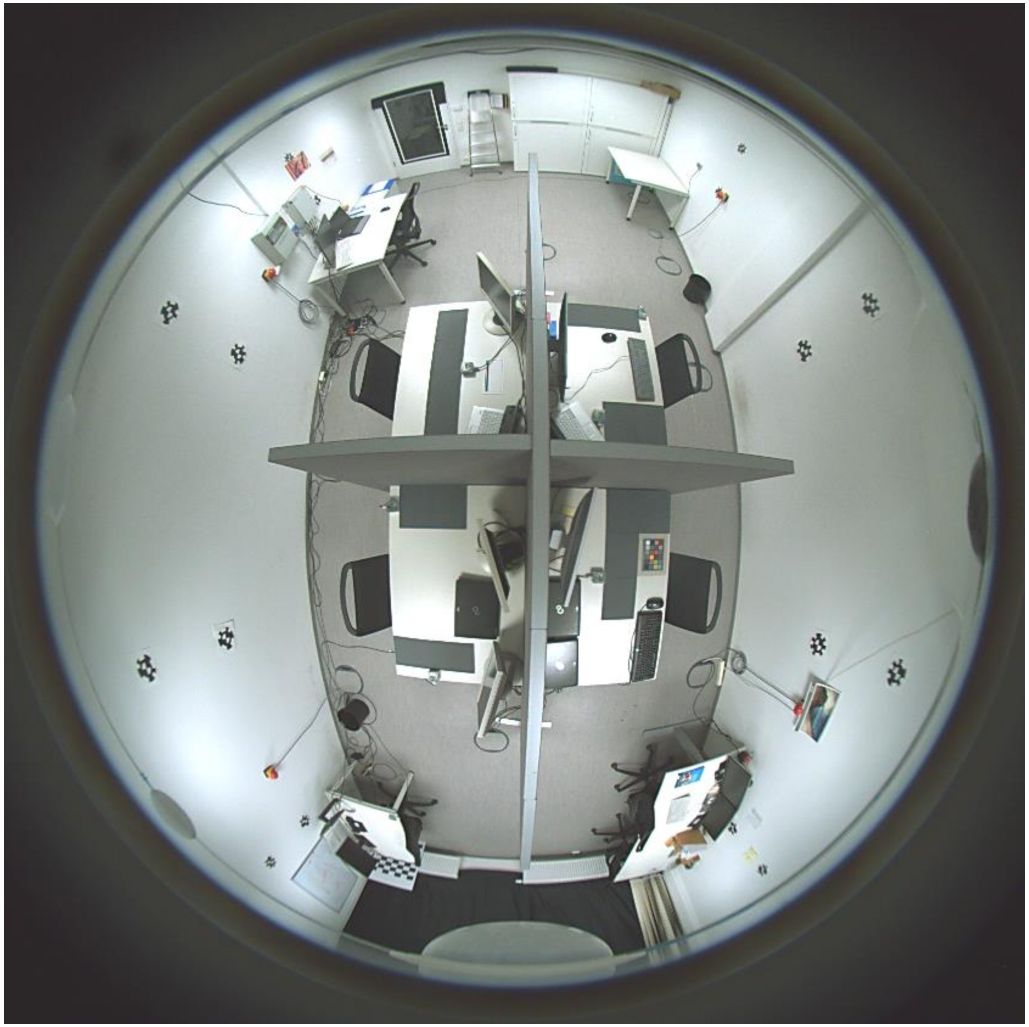
Radiosity solution

$$\begin{bmatrix} 1-\rho_1 F_{11} & -\rho_1 F_{12} & \cdots & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1-\rho_1 F_{22} & \cdots & -\rho_2 F_{2n} \\ \vdots & \vdots & \ddots & \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \cdots & 1-\rho_1 F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{bmatrix}$$

- ❑ “full matrix” radiosity solution
  - ❑ Solves simultaneous linear equations
  - ❑ Complete solution all together
  - ❑ Expensive in time (number of faces, complex scenes tend to be ten of thousand faces)
  - ❑ Expensive to store in memory

- ❑ “Progressive” radiosity solution
  - ❑ incremental method, solves it iteratively
  - ❑ yielding intermediate results at much lower computation and storage costs
  - ❑ update the scene each time, user can even stop the iteration in case he believes that the output is realistic enough without waiting for convergence

Real life…

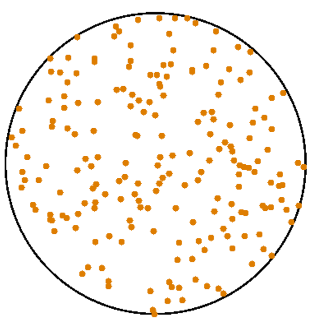


Form factors

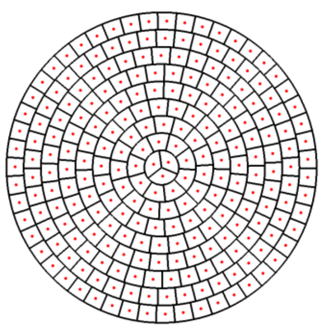
- Ray tracing problem → fraction of rays (light energy) arriving at patch j, *mj*, from the total rays leaving patch i, , *mi*

$$f_{ij} = \frac{m_j}{m_i}$$

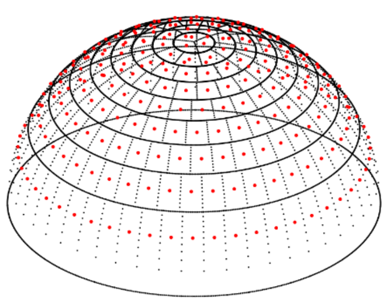
- Enhances the uniformity of the generated quasi-random sequence of ray directions and leads to faster convergence



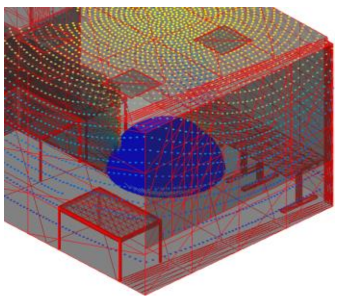
Monte carlo



Isocell
(unit disc)

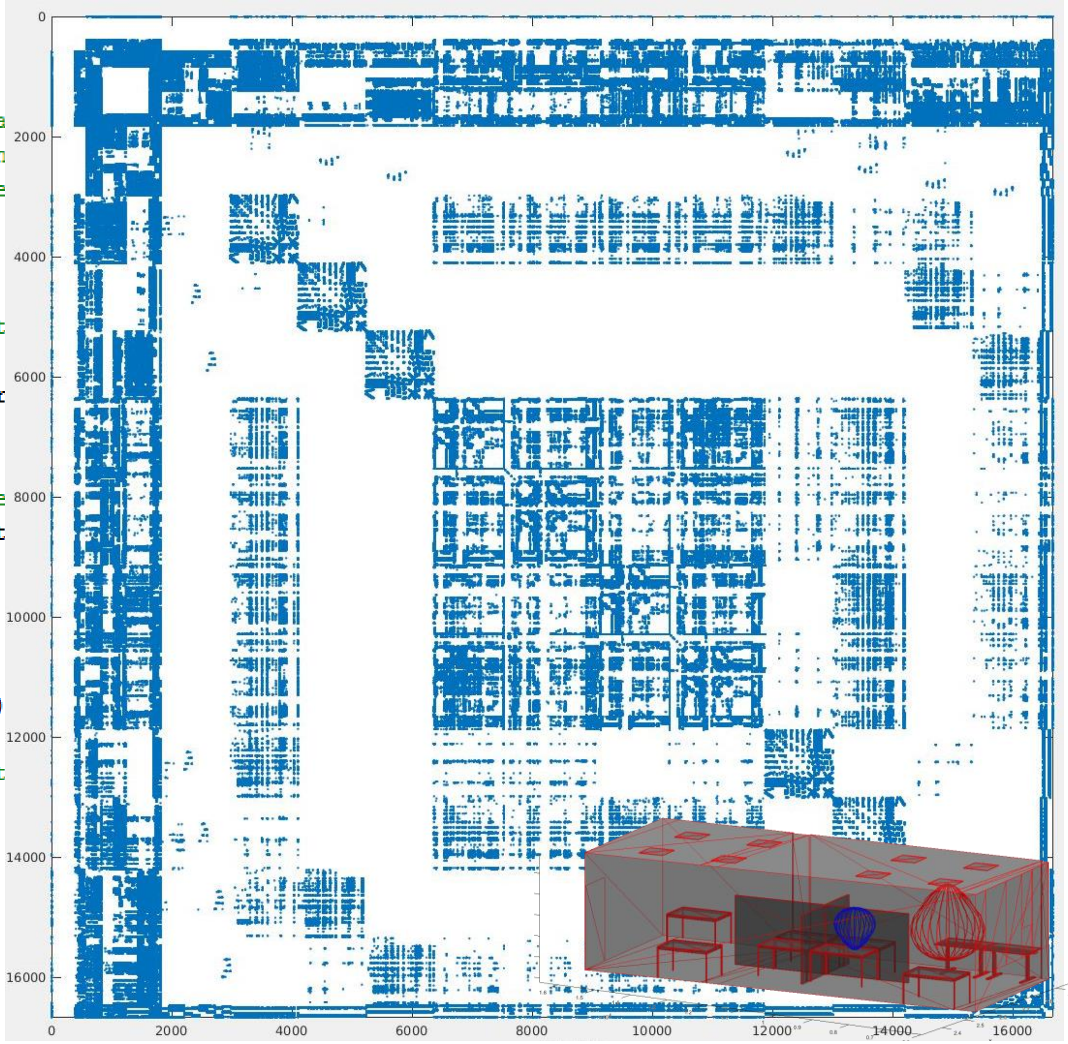


Isocell
(unit sphere)



Form factors in real scenarios

```
t = opcodemesh(v',f');
for i = patches;
    r = vrrotvec(normals(i,:),[0 0 1]); % Calculate rota
    M = vrrotvec2mat(r); % Convert rotation from axis-an
    draysR = drays*M; % coincident rays to the normal ve



    % start throwing rays in space and finding intersect
    % other pactches/faces
    [hit,tt,idxx,bary,xnodee] = t.intersect(repmat(centr
    xnodee = xnodee';

    intersections = unique(idxx); % find which faces are
    intersections(isnan(intersections) | isinf(intersect
    intersections(intersections == 0) = [];

    if ~isempty(intersections)

        ind22 = ~isFacing(centroids(i, :), normals(i, :)

        % and remove them since they do not face each ot
        intersections(ind22) = [];
    end
    ...
    ...
    ...
```

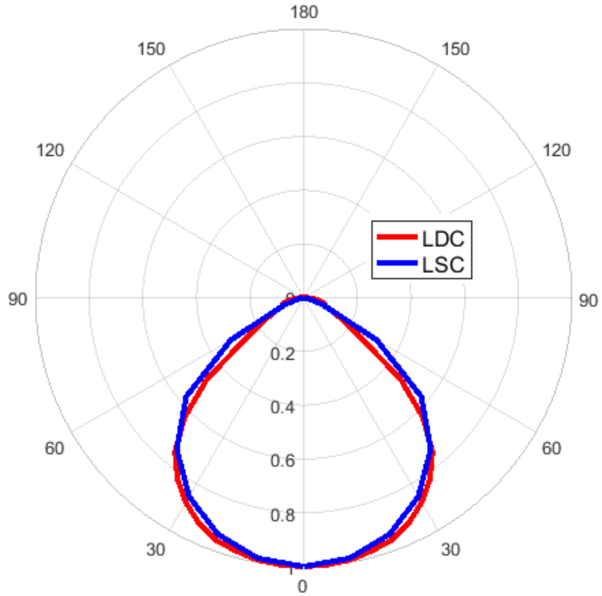


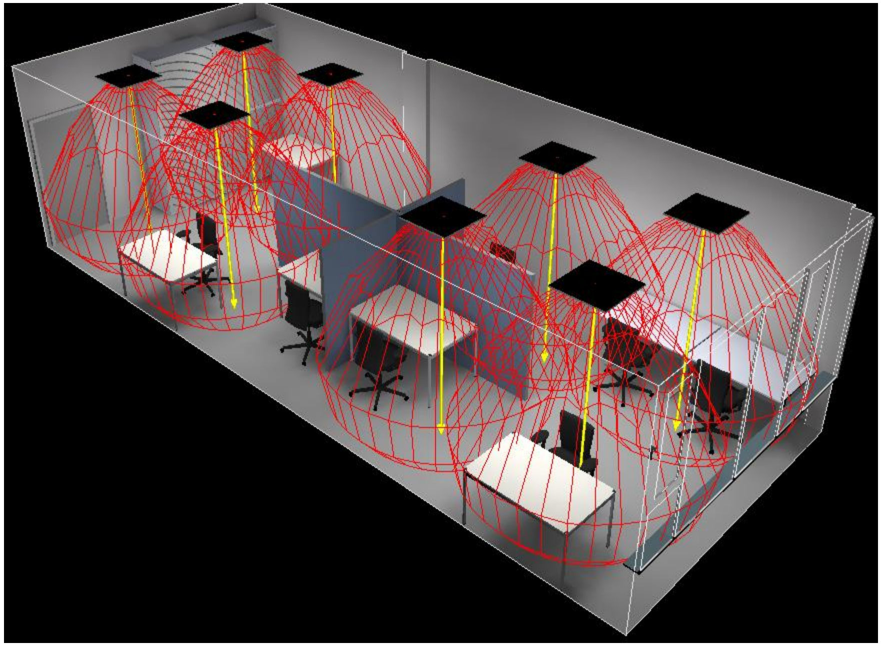
3DV 2018 SceneUnderLight European Commission iit ISTITUTO ITALIANO DI TECNOLOGIA OSRAM

Light sources and light perception

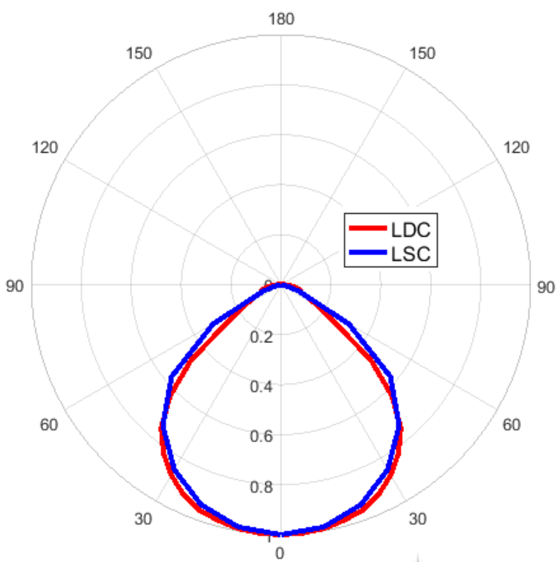
- ❑ Point based isotropic light sources vs. Luminaires
  - ❑ Luminous intensity (lumens)
  - ❑ Light Distribution Curve (LDC)

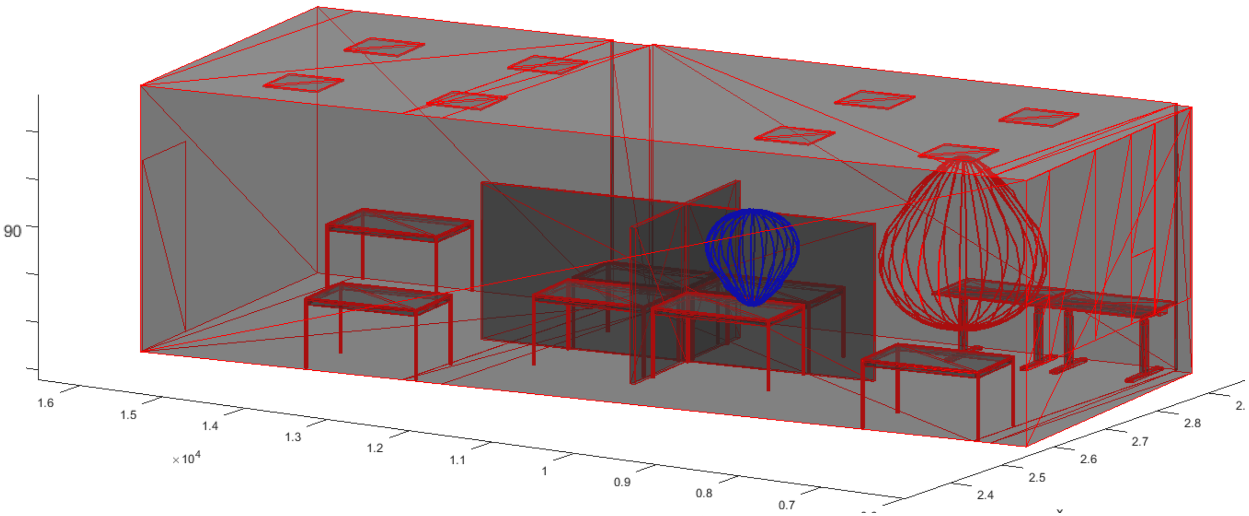
- ❑ Disregards light perception
  - ❑ Luxmeter's Sensitivity Curve (LSC)

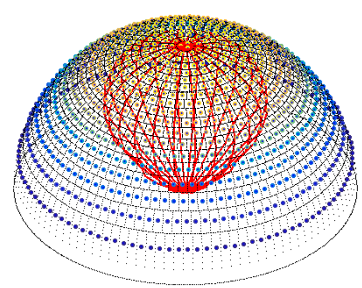




LDC, LSC





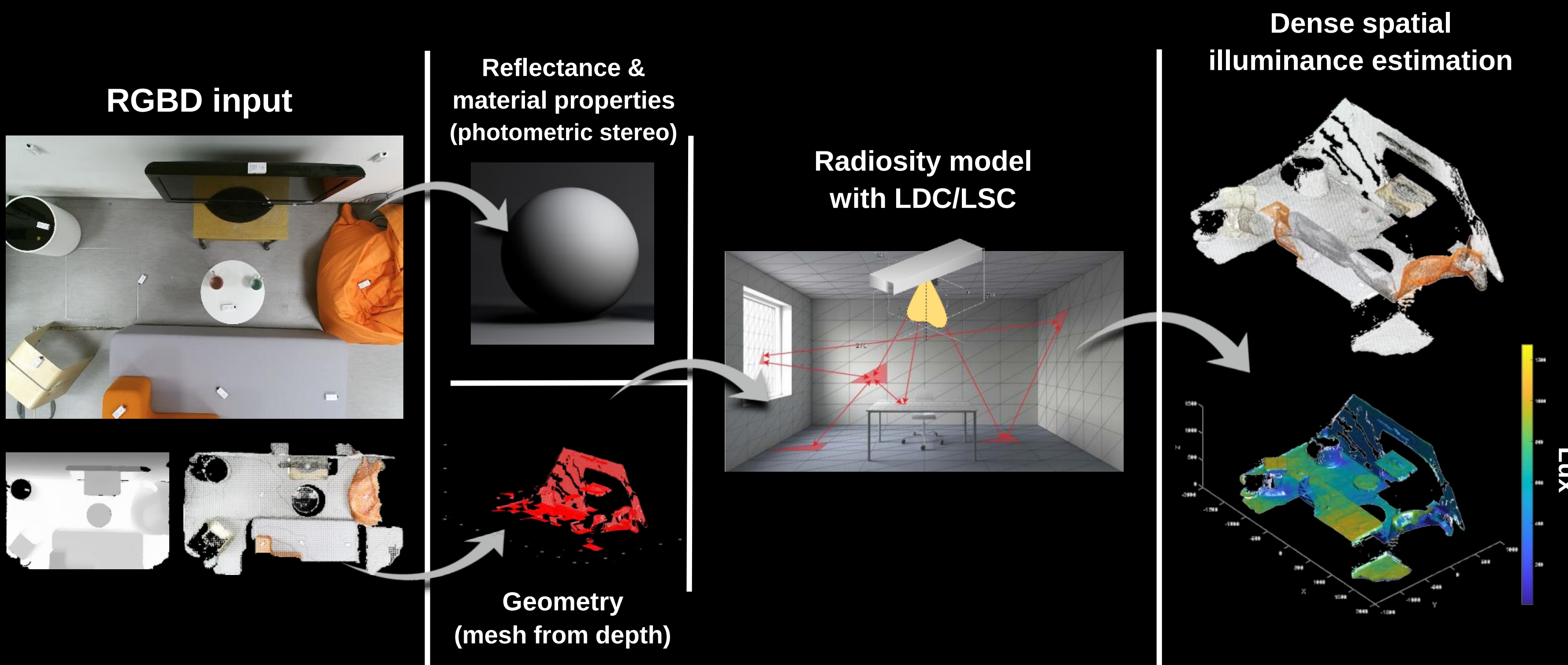


Weighted isocell

(unit sphere, LDC/LSC)

$$f_{ij} = \frac{m_j}{m_i}$$

Is a camera-aided rendering technique sufficient enough for light modeling?



Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

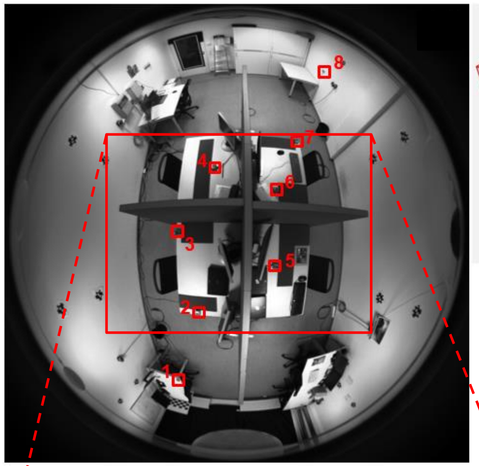
Practical examples

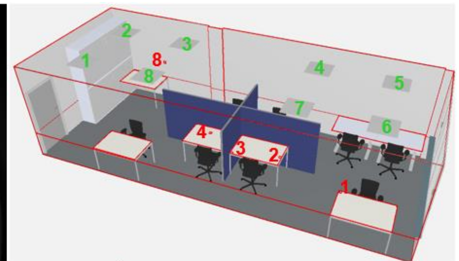
Applications

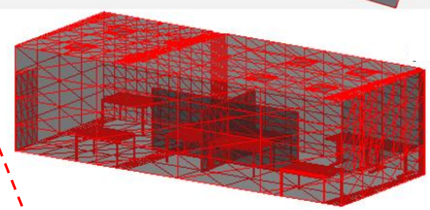
Conclusion & future work

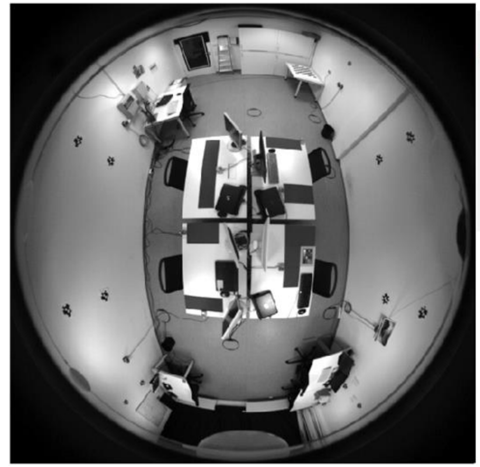
Dataset

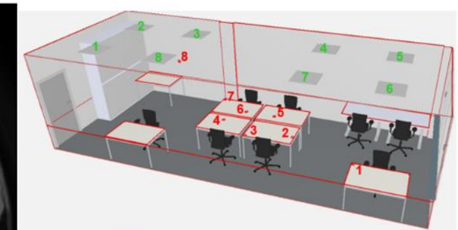
Room 1 | Room 2

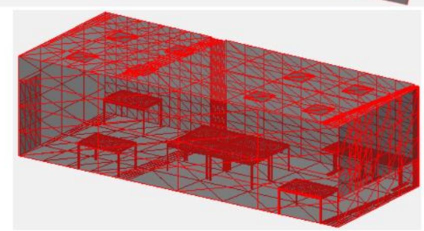




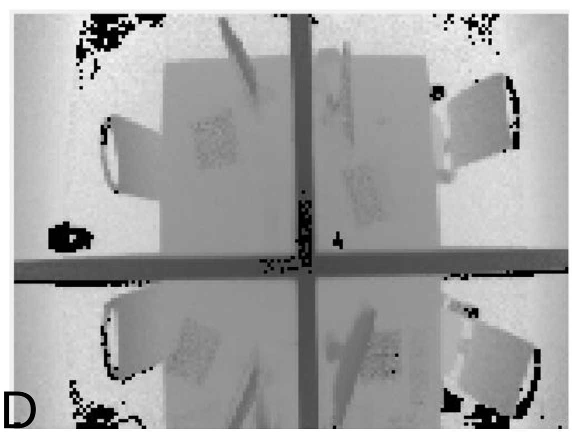


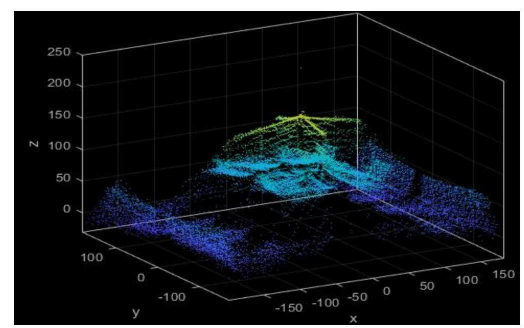






RGBD



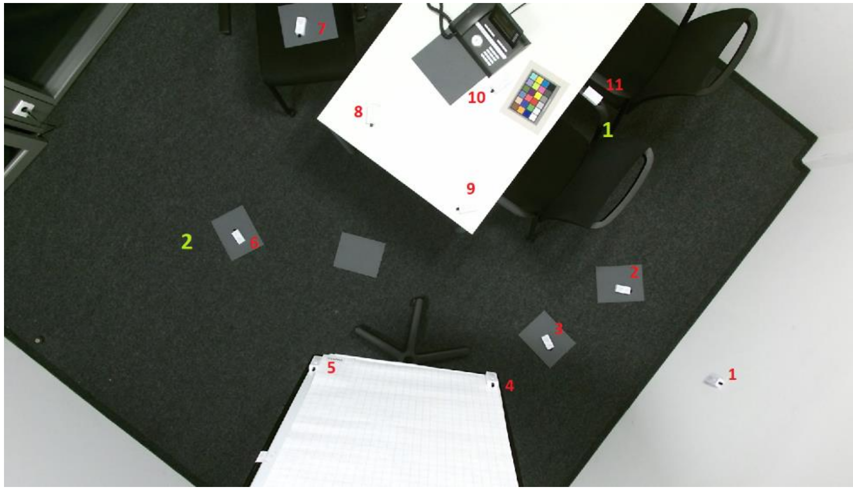


Point cloud

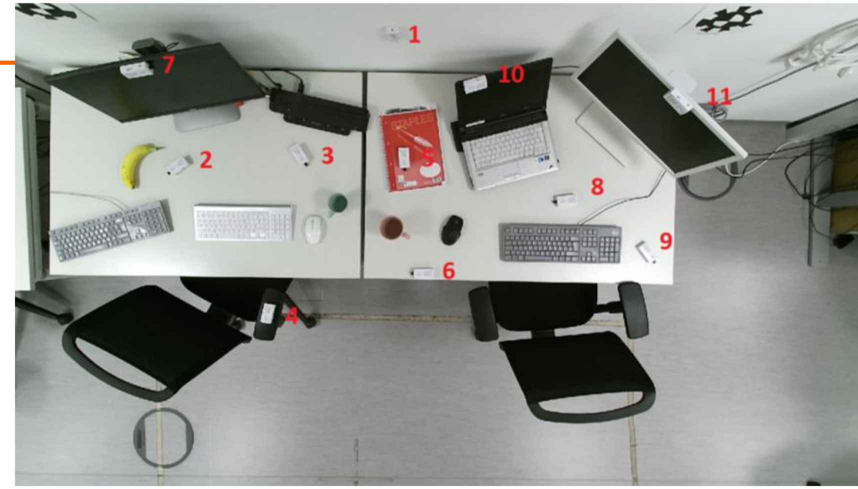
3DV 2018 SceneUnderLight European Commission IIT VGM ISTITUTO ITALIANO DI TECNOLOGIA OSRAM

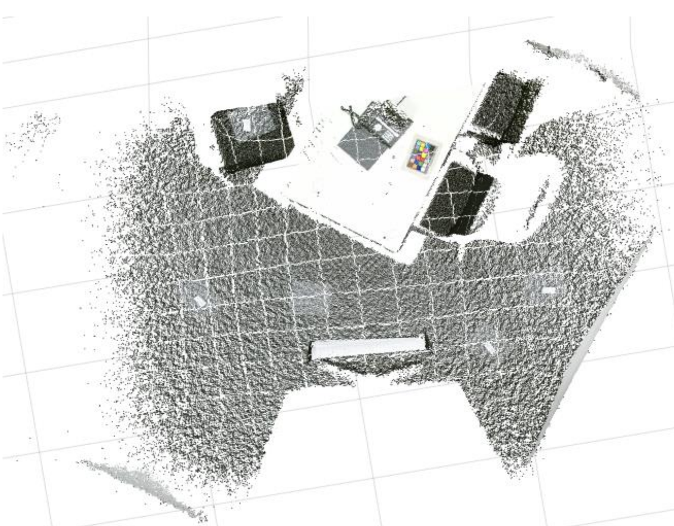
Dataset

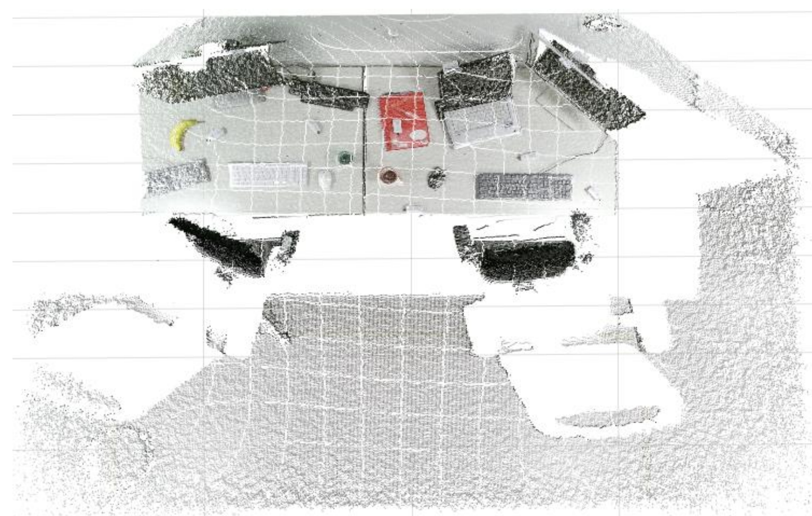
Room 3



Room 4

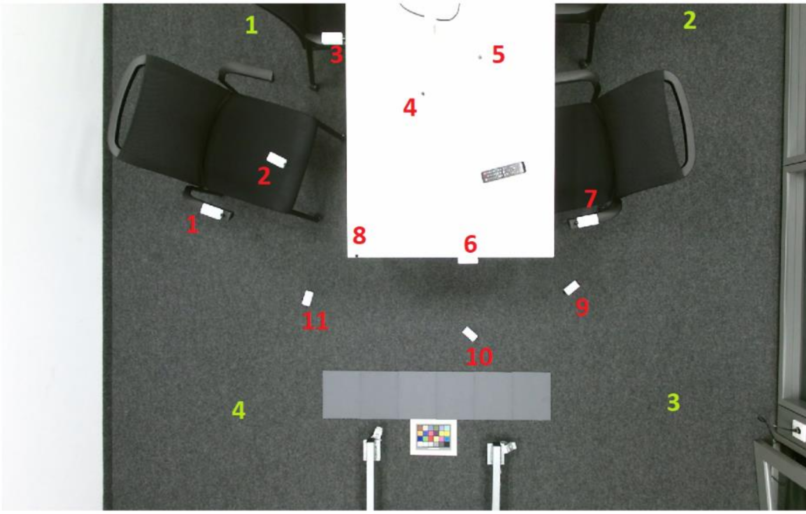




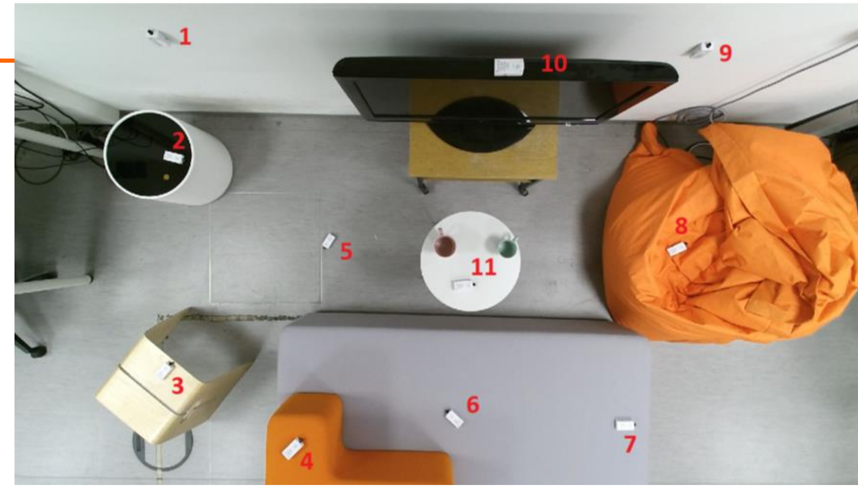


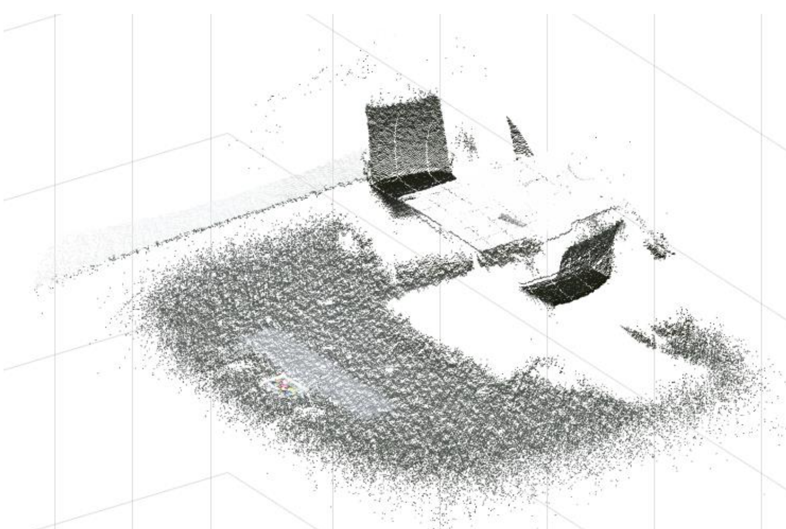
Dataset

Room 5



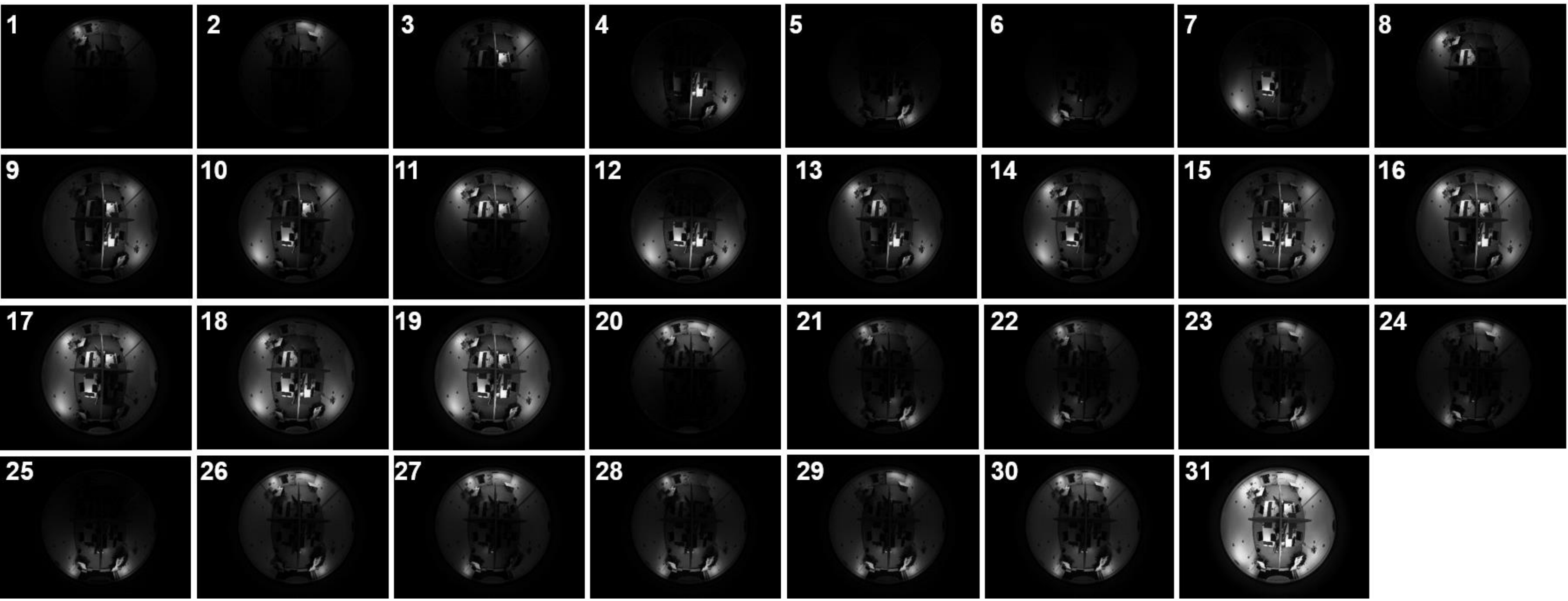
Room 6





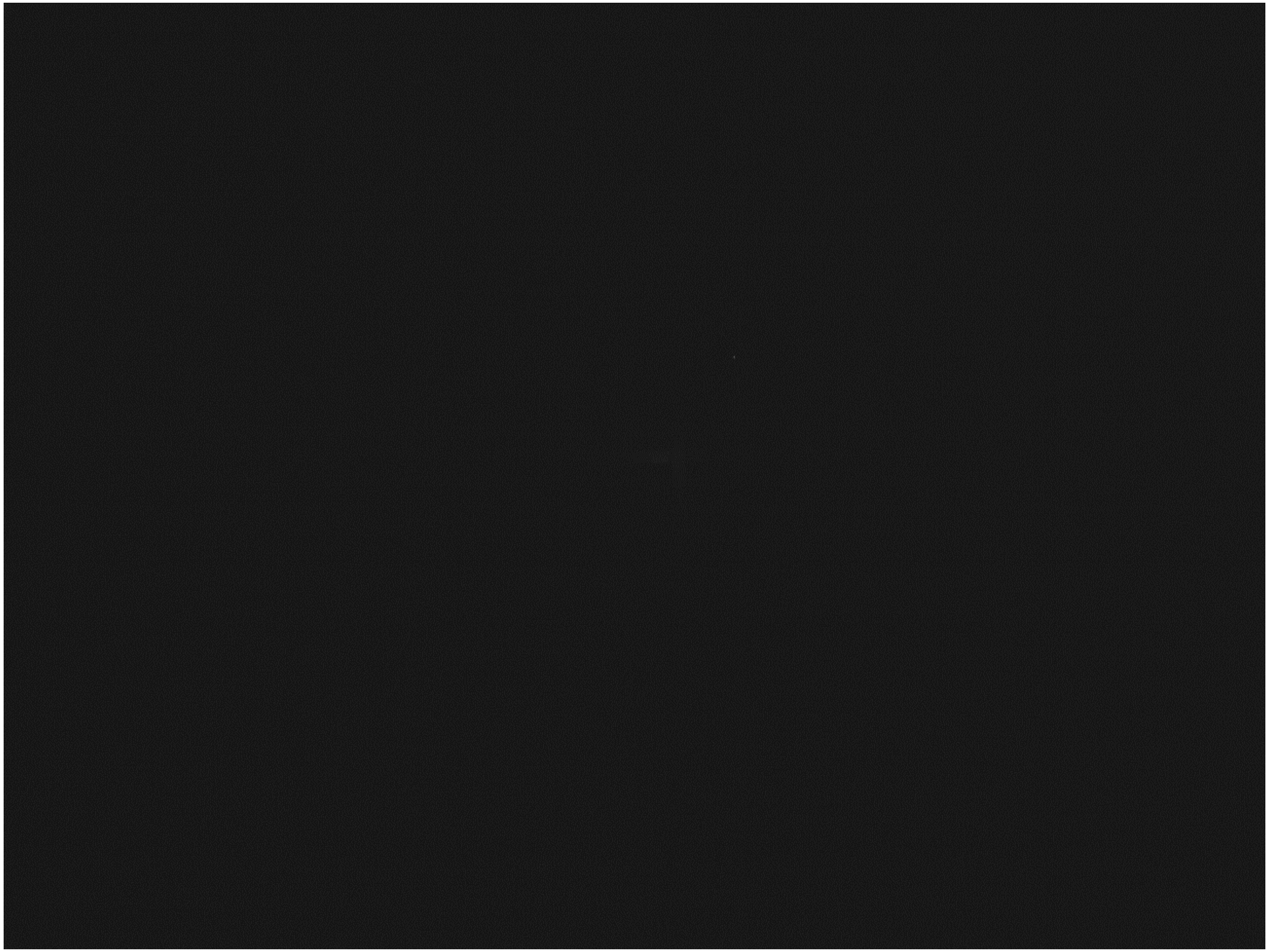
Dataset

(Illumination combinations)



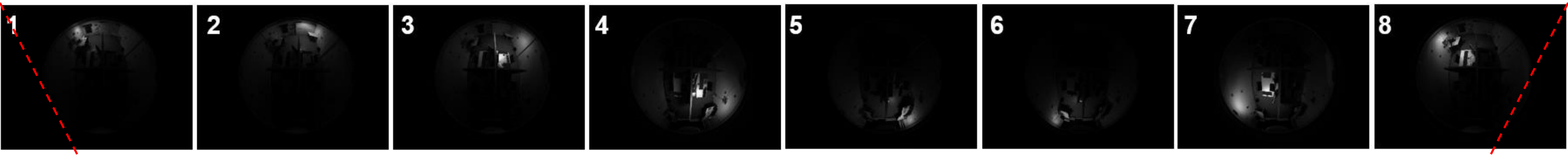
Dataset

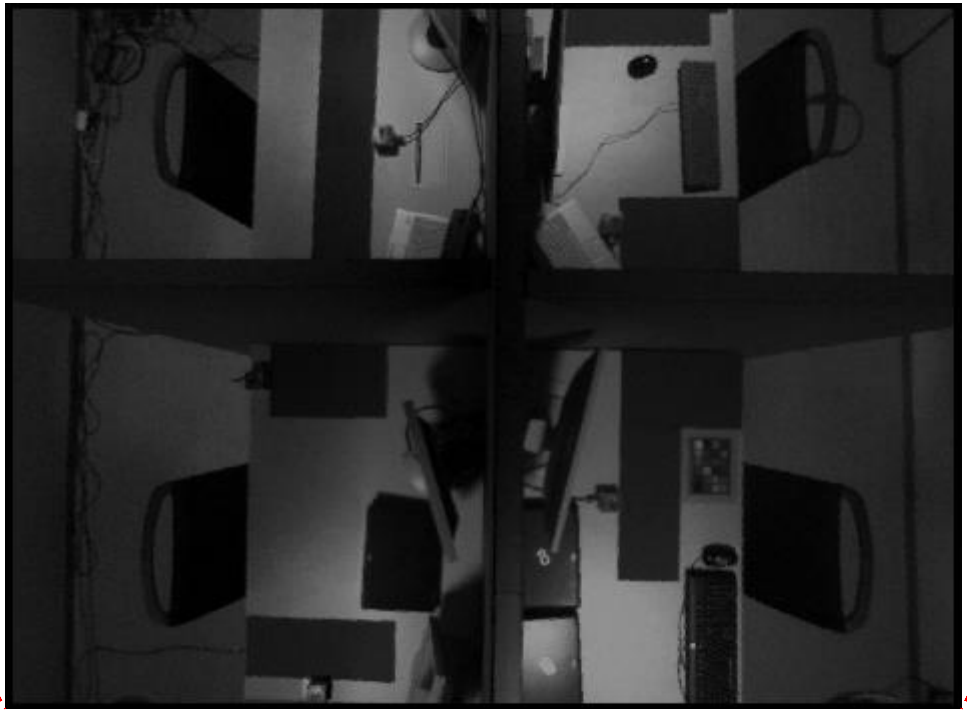
(Illumination combinations)



Albedo – photometric stereo

Individual light sources

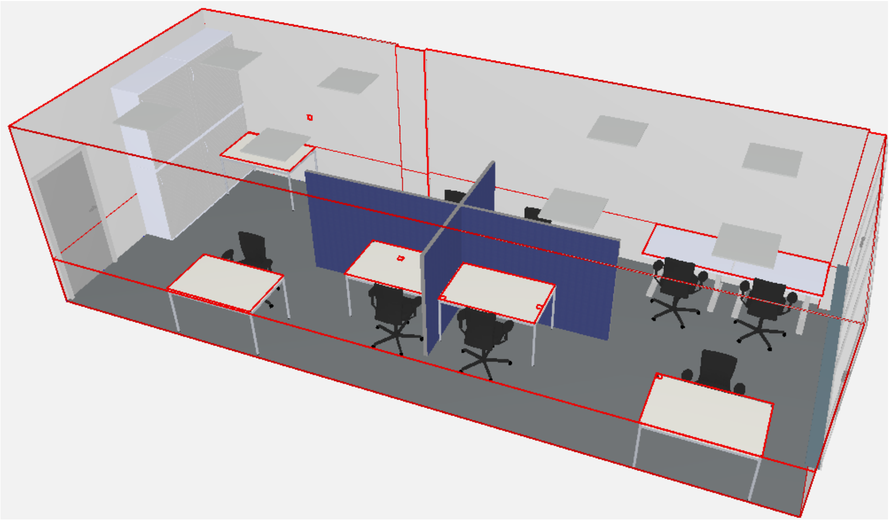




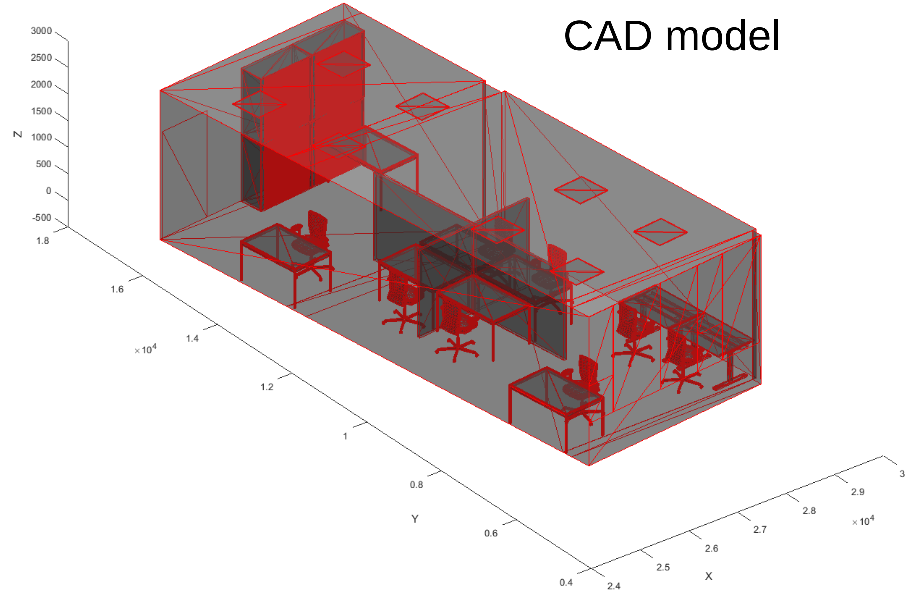
Albedo map

Ablation studies

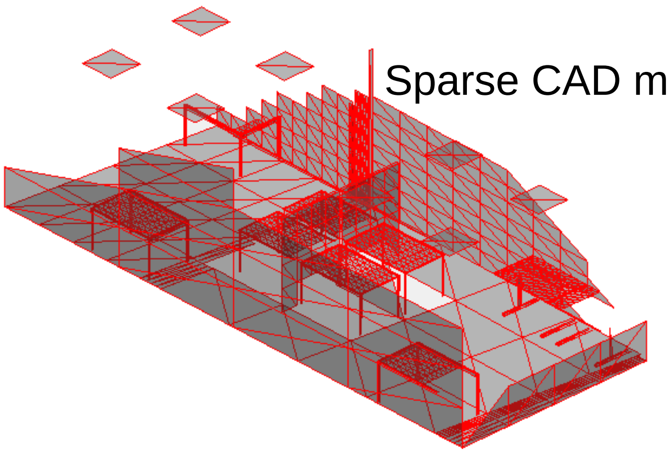
CAD model
(Relux)



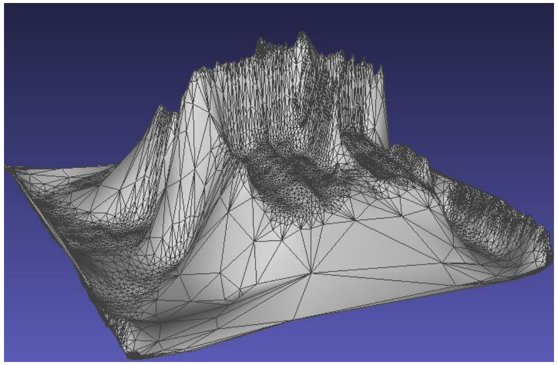
CAD model



Sparse CAD model



Mesh



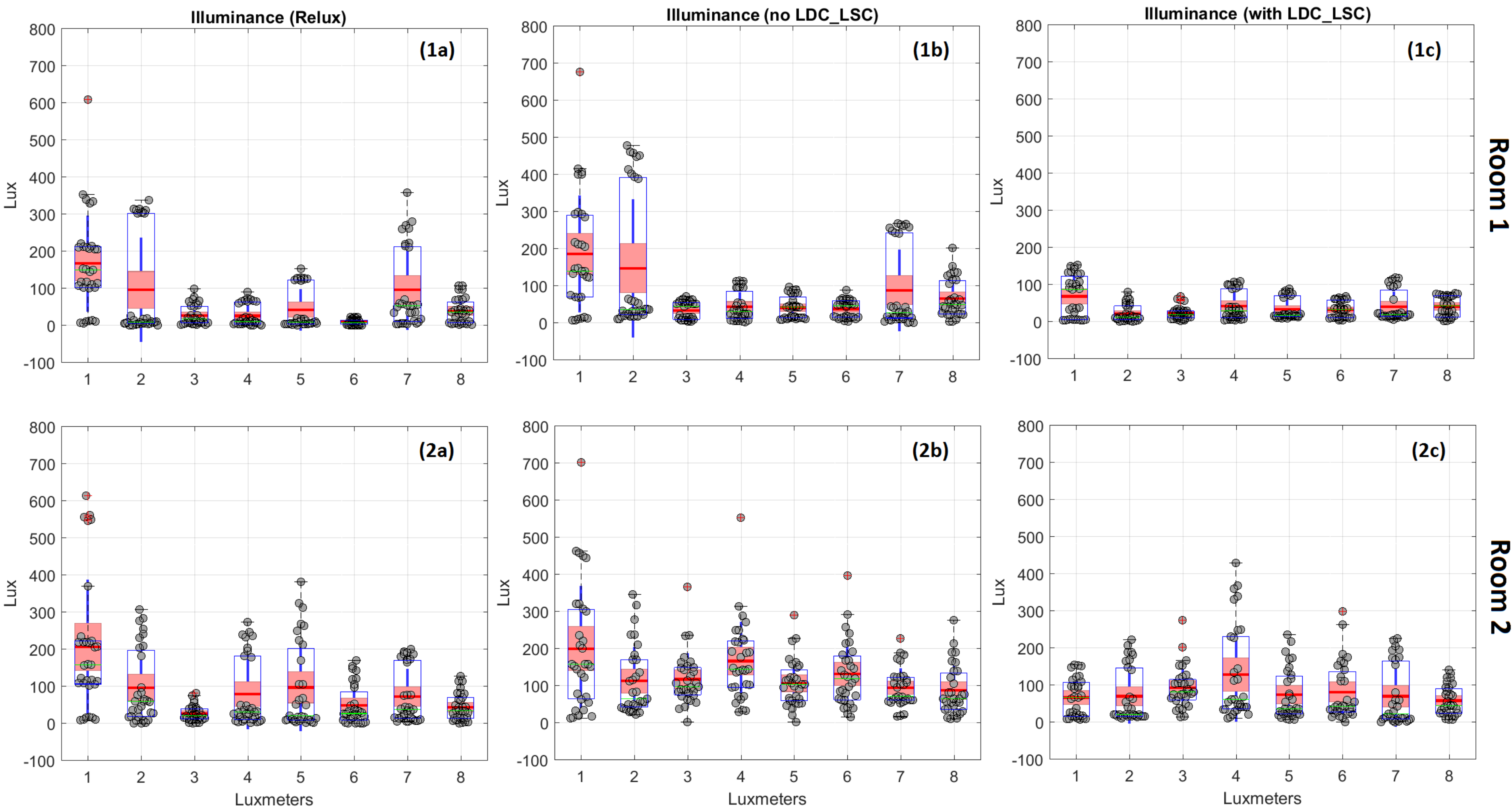
3DV 2018 SceneUnderLight European Commission iit ISTITUTO ITALIANO DI TECNOLOGIA OSRAM

Results

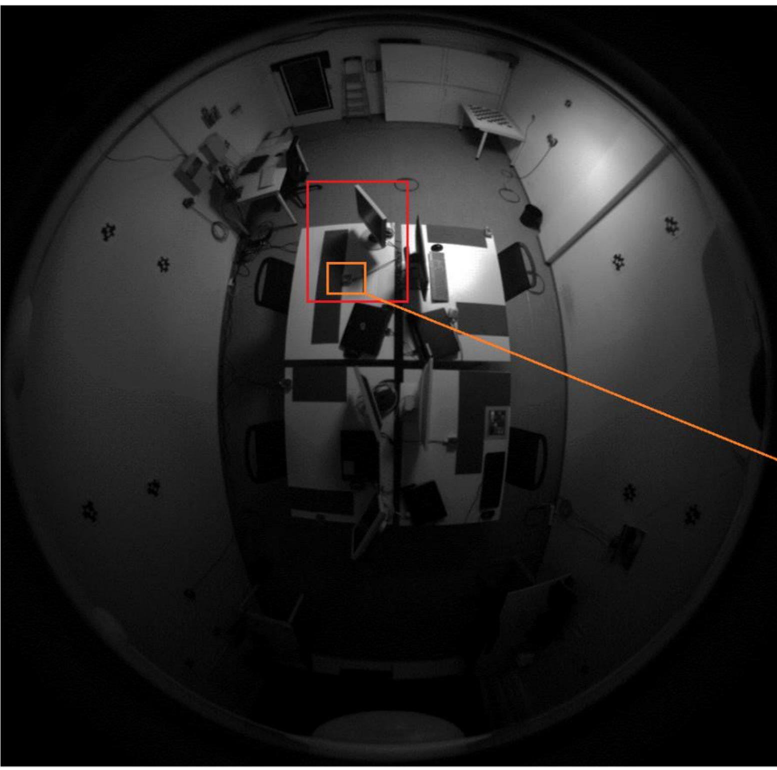
	Room 1 Error (in Lux)									
	Luxmeters									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Avg. (1-8)**	**Avg. (2-7)**
Relux	167	96	27	26	43	10	96	39	63 (21.4%)	50 (20.7%)
Ours with CAD (no_LDC_LSC)	188	150	33	45	43	34	91	65	81 (27.5%)	66 (27.3%)
Ours with CAD (LDC)	199	152	29	41	40	33	95	57	81 (27.5%)	65 (26.9%)
Ours with CAD (LSC)	73	45	24	32	40	34	46	52	43 (14.6%)	37 (15.3%)
Ours with CAD (LDC_LSC)	69	24	22	38	28	28	38	41	**36 (12.2%)**	**30 (12.4%)**
Ours with CAD Camera visible (LDC_LSC)	-	64	28	20	17	22	52	-	-	34 (14.1%)
Ours RGB2Lux (LDC_LSC)	-	53	41	67	68	40	98	-	-	61 (25.3%)

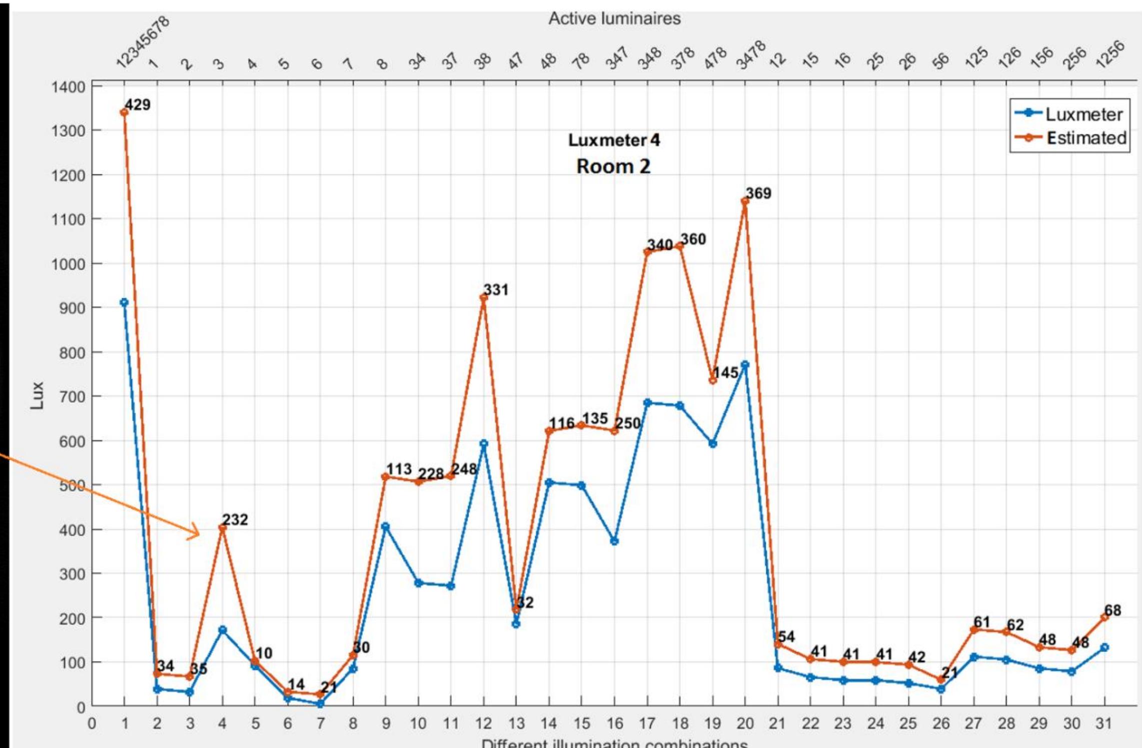
	Room 2 Error (in Lux)									
	Luxmeters									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Avg. (1-8)**	**Avg. (2-7)**
Relux	206	97	27	80	97	49	73	44	84 (22.2%)	71 (20.4%)
Ours with CAD (no_LDC_LSC)	207	114	99	148	105	117	93	81	120 (31.8%)	112 (32.2%)
Ours with CAD (LDC)	213	117	82	125	97	97	86	63	110 (29.1%)	100 (28.8%)
Ours with CAD (LSC)	69	80	98	136	70	84	56	62	82 (21.7%)	87 (25.0%)
Ours with CAD (LDC_LSC)	70	57	76	106	75	69	55	53	**70 (18.5%)**	73 (21.0%)
Ours with CAD Camera visible (LDC_LSC)	-	54	36	59	101	69	54	-	-	**62 (17.8%)**
Ours RGB2Lux (LDC_LSC)	-	98	90	85	136	108	77	-	-	99 (28.5%)

Results – CAD model

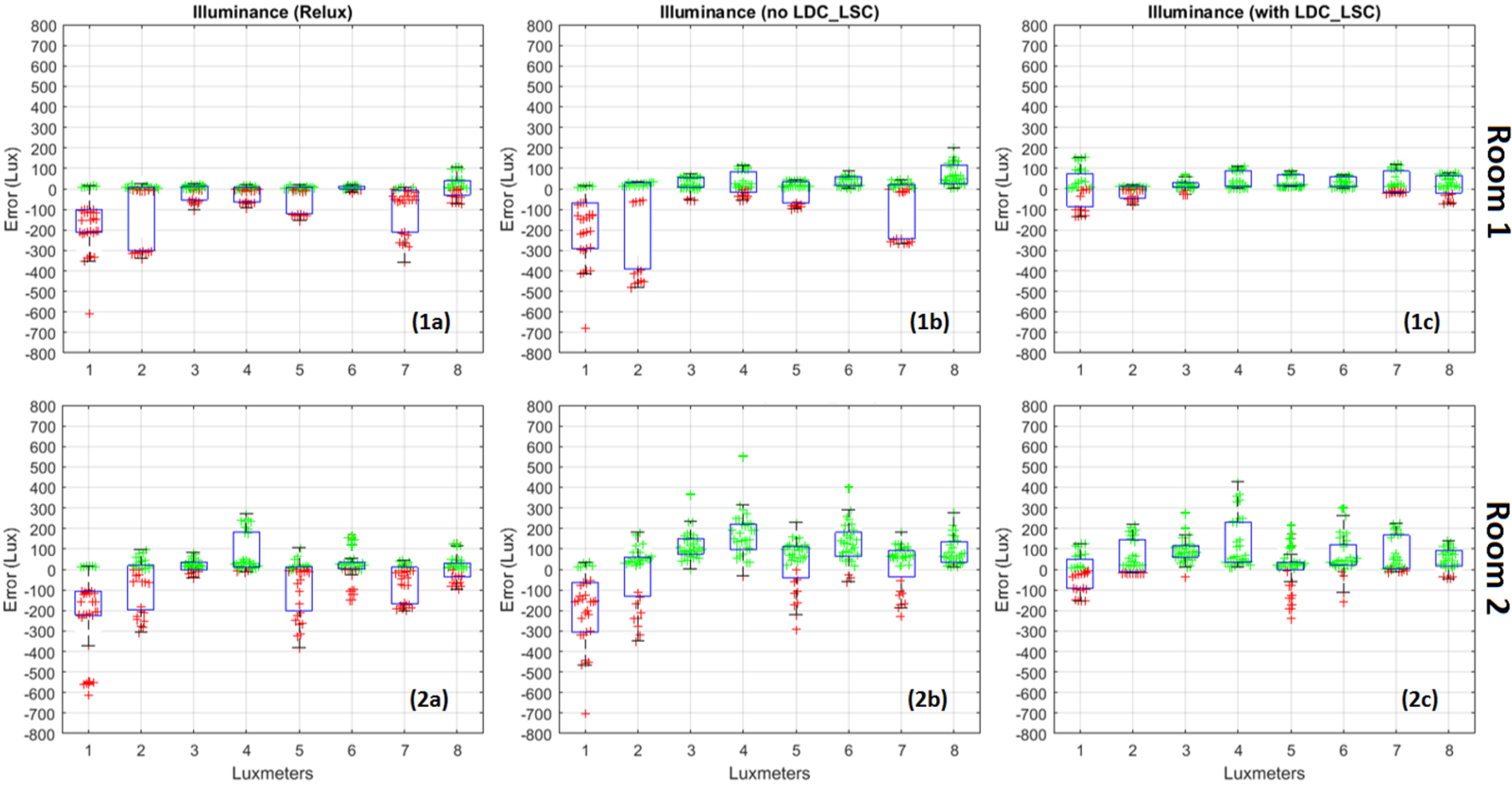


Results – Luxmeter 4, room 2

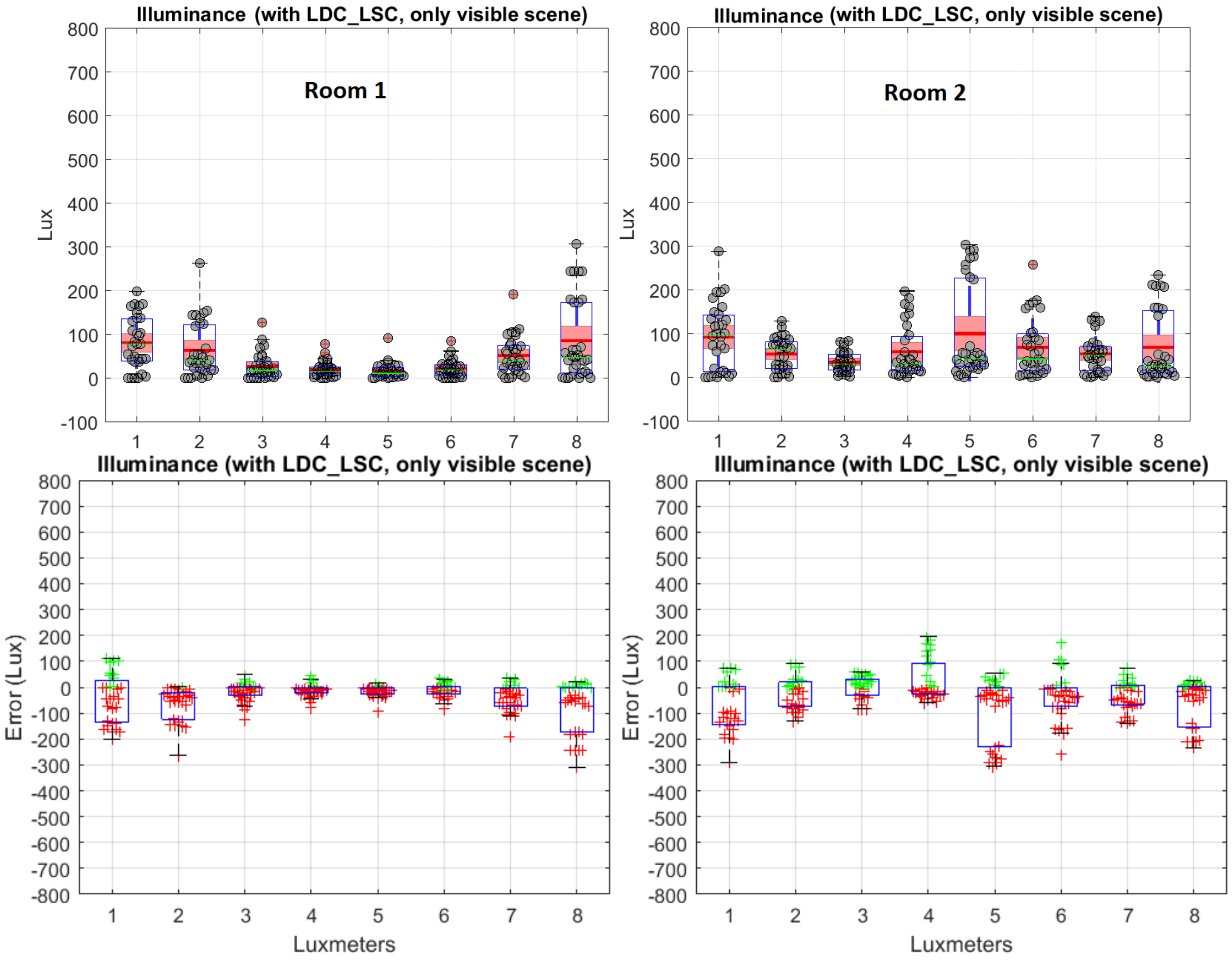




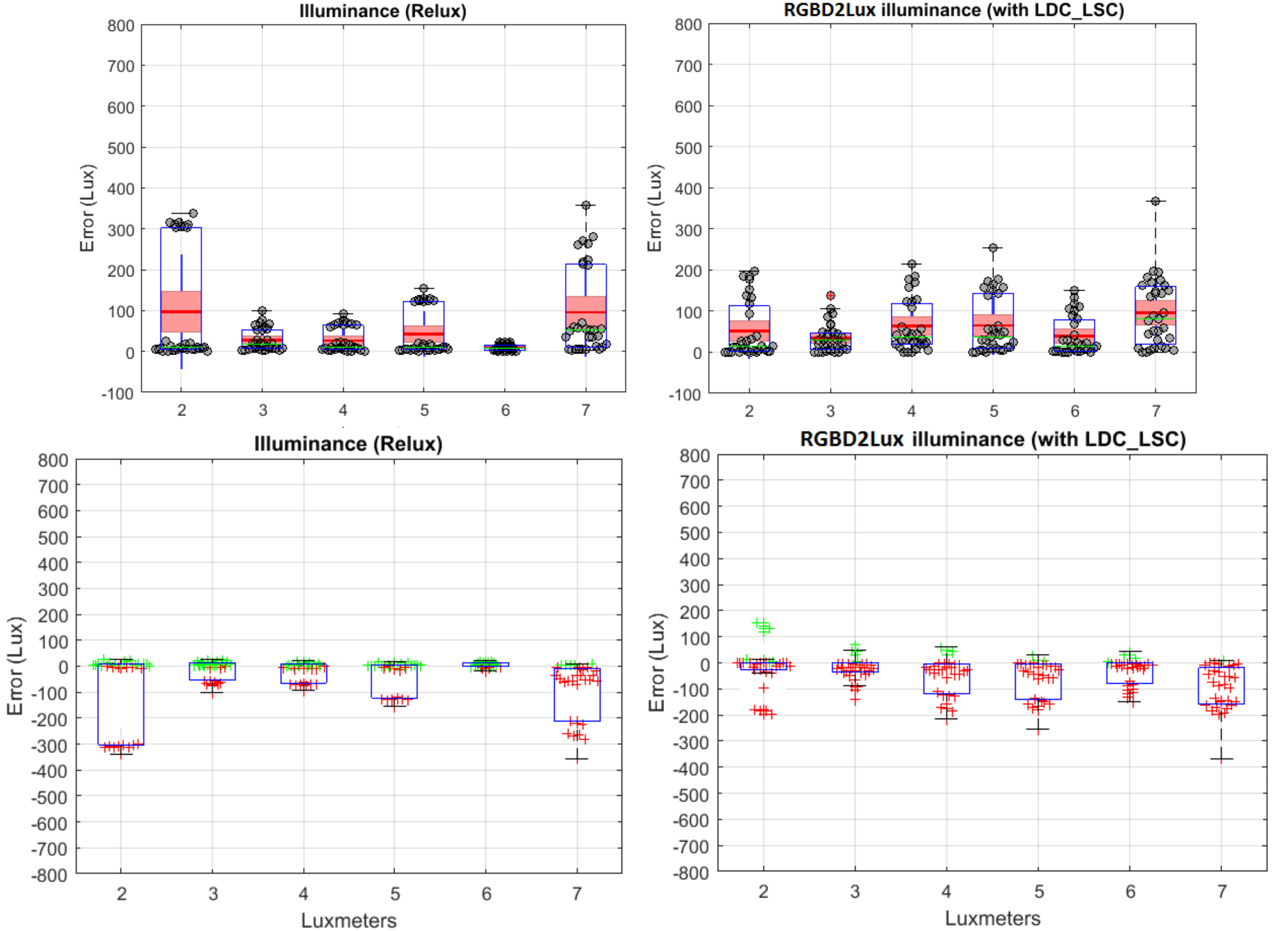
Results – CAD model
(under/over estimation)



Results – partial CAD model

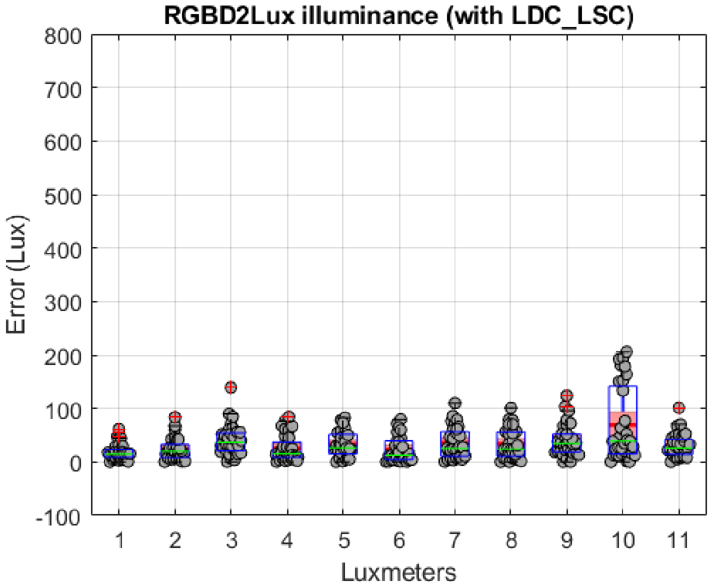


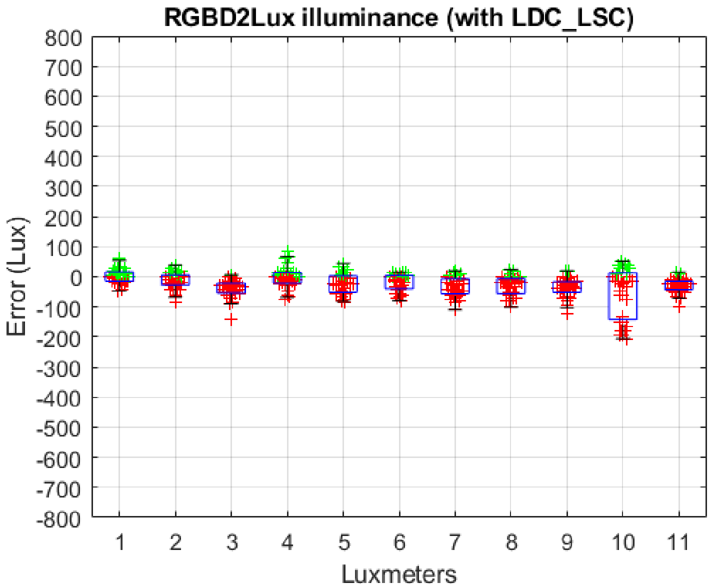
Results – 3D mesh model (Room1)

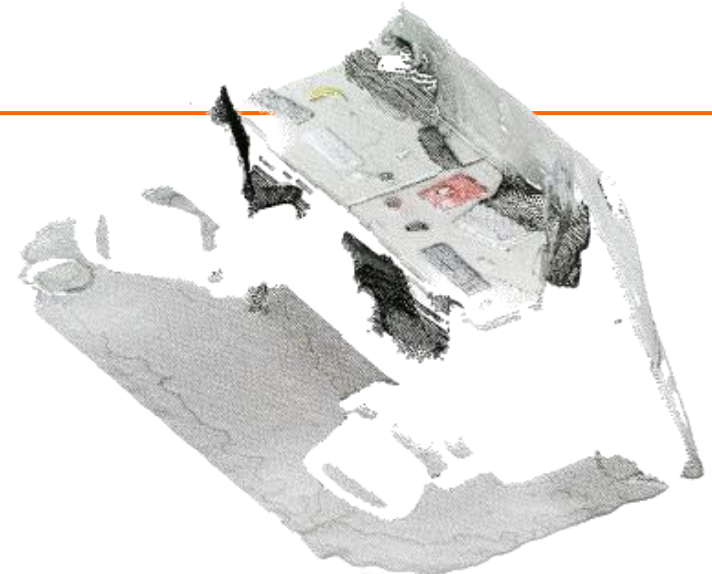


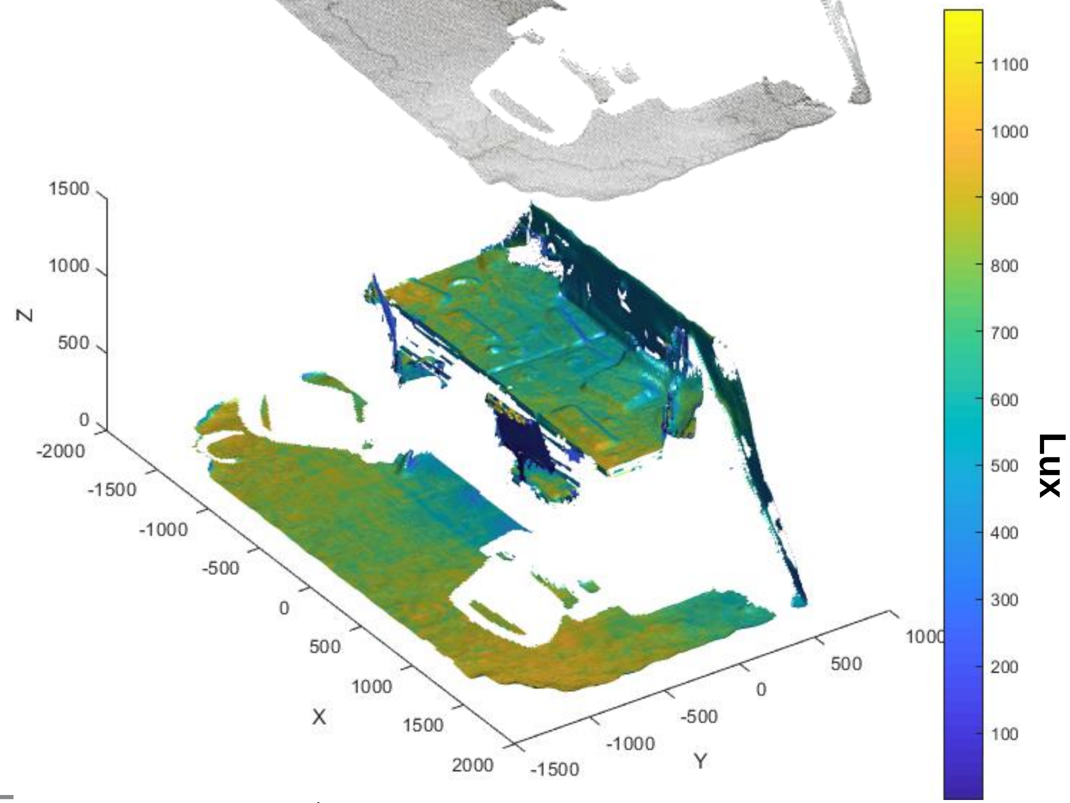
3DV 2018
SceneUnderLight
European Commission
iit
ISTITUTO ITALIANO DI TECNOLOGIA
OSRAM

Results – 3D mesh model (Room4)

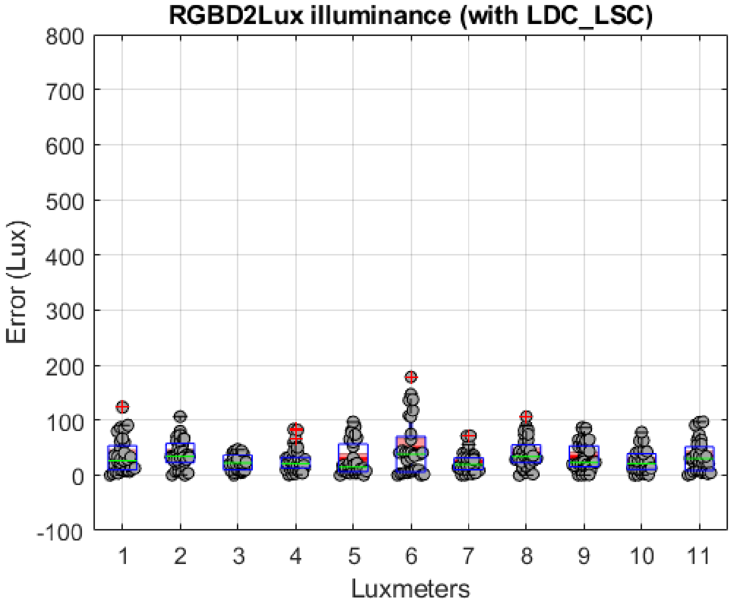


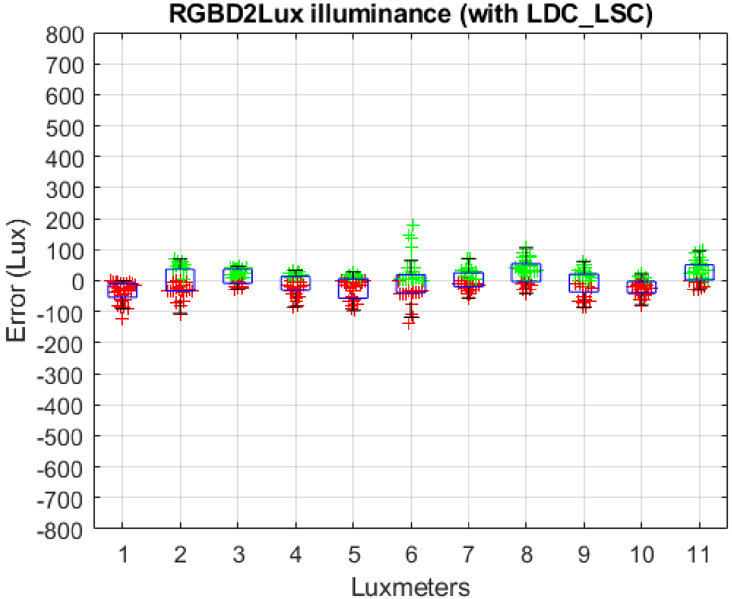


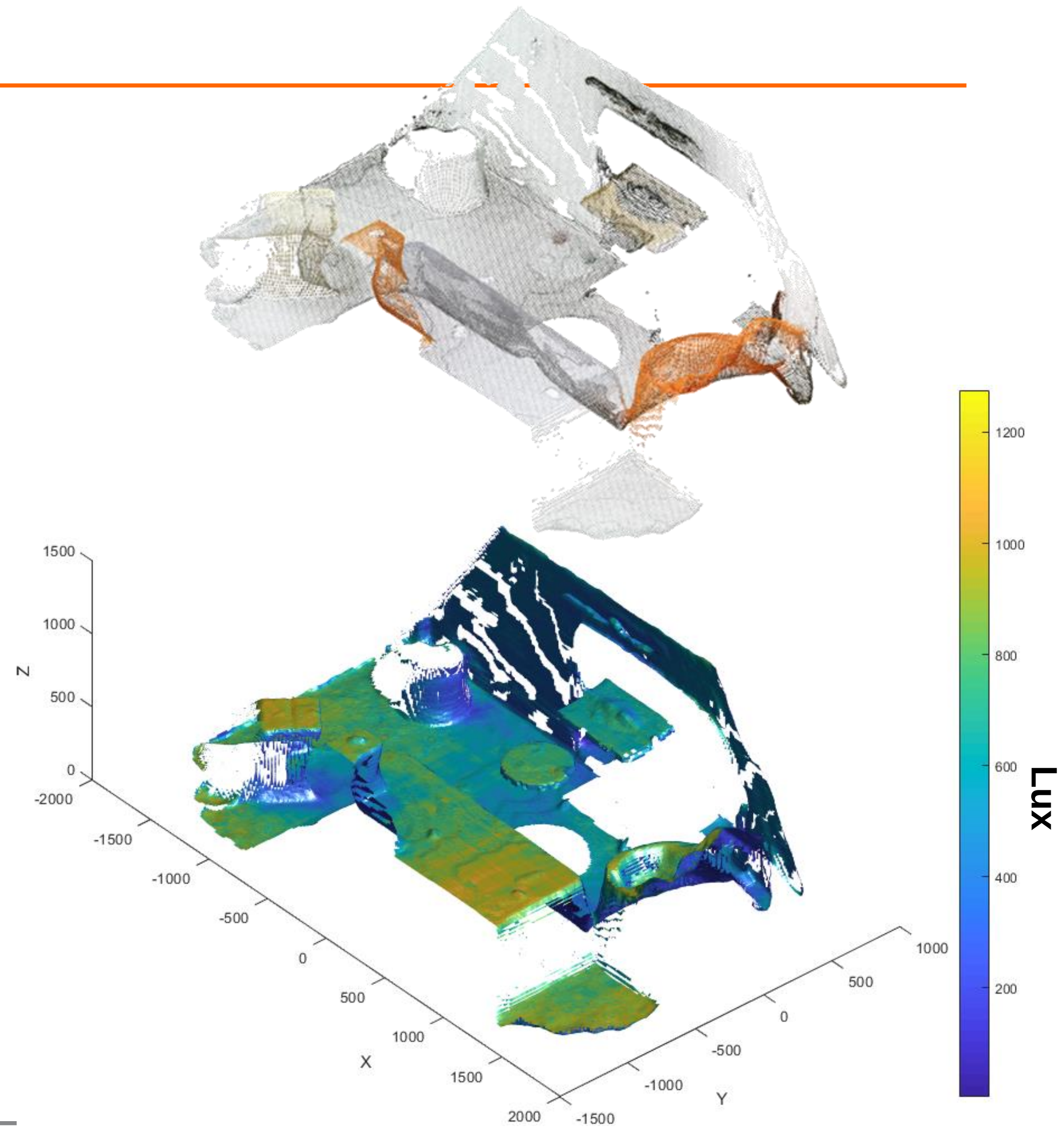




Results – 3D mesh model (Room6)







3DV 2018 · SceneUnderLight · European Commission · IIT Istituto Italiano di Tecnologia VGM · OSRAM

Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

Practical examples

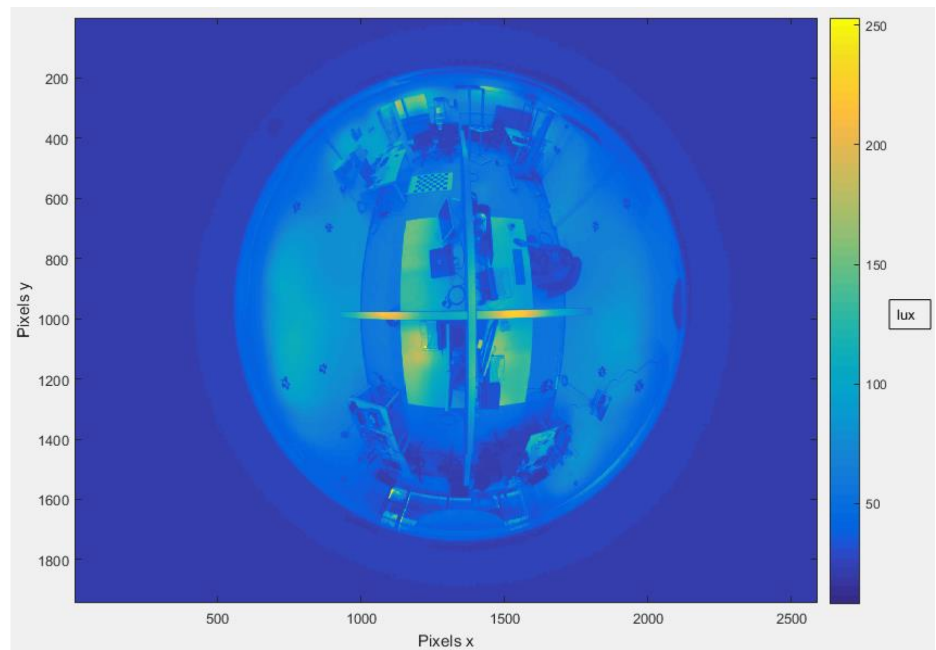
Applications

Conclusion & future work

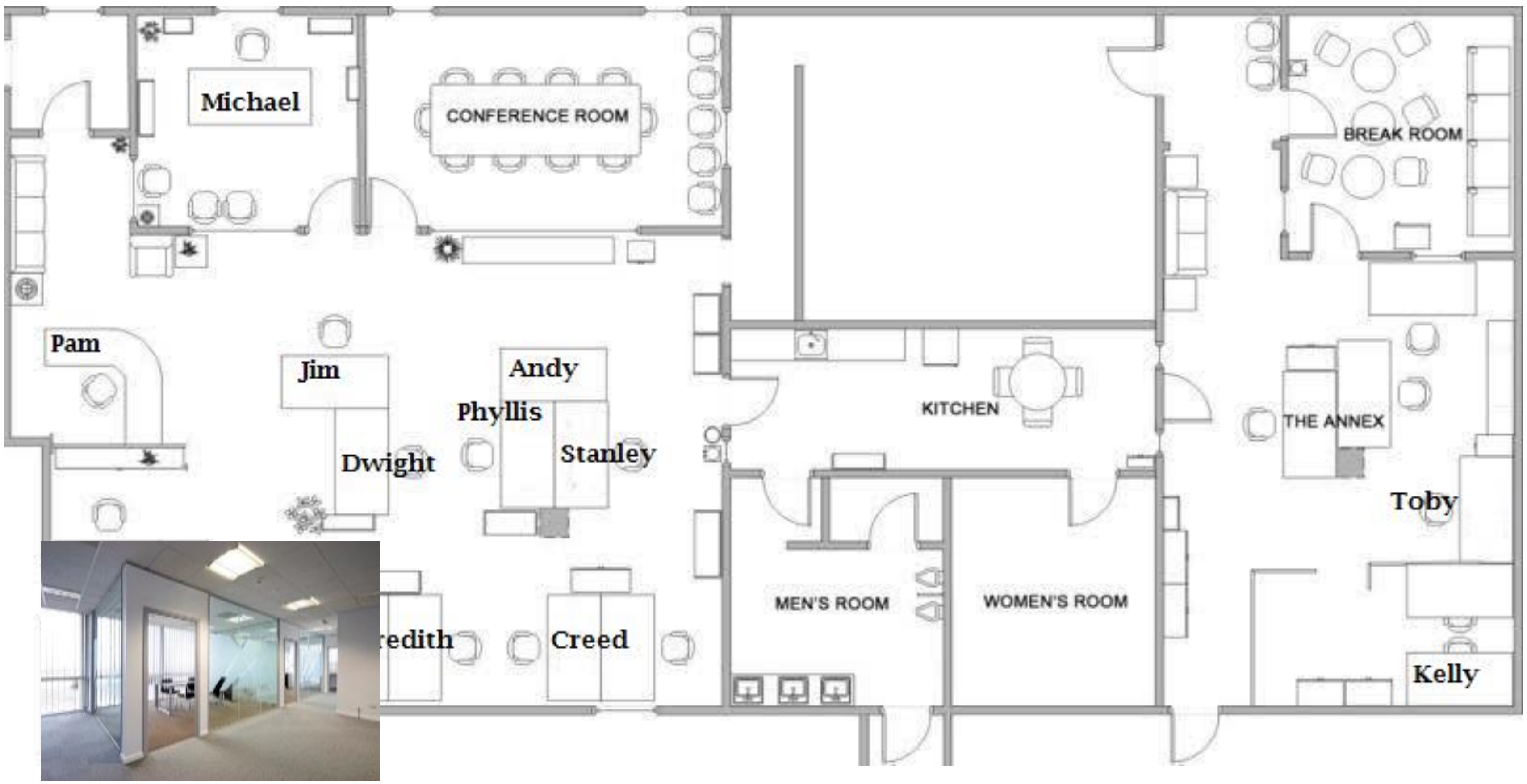
Estimate dense spatial illumination

- ❑ Knowledge of spatial illumination over time
  - ❑ Light commissioning
  - ❑ Smart lighting management systems
- ❑ Camera input --> map pixels to lux

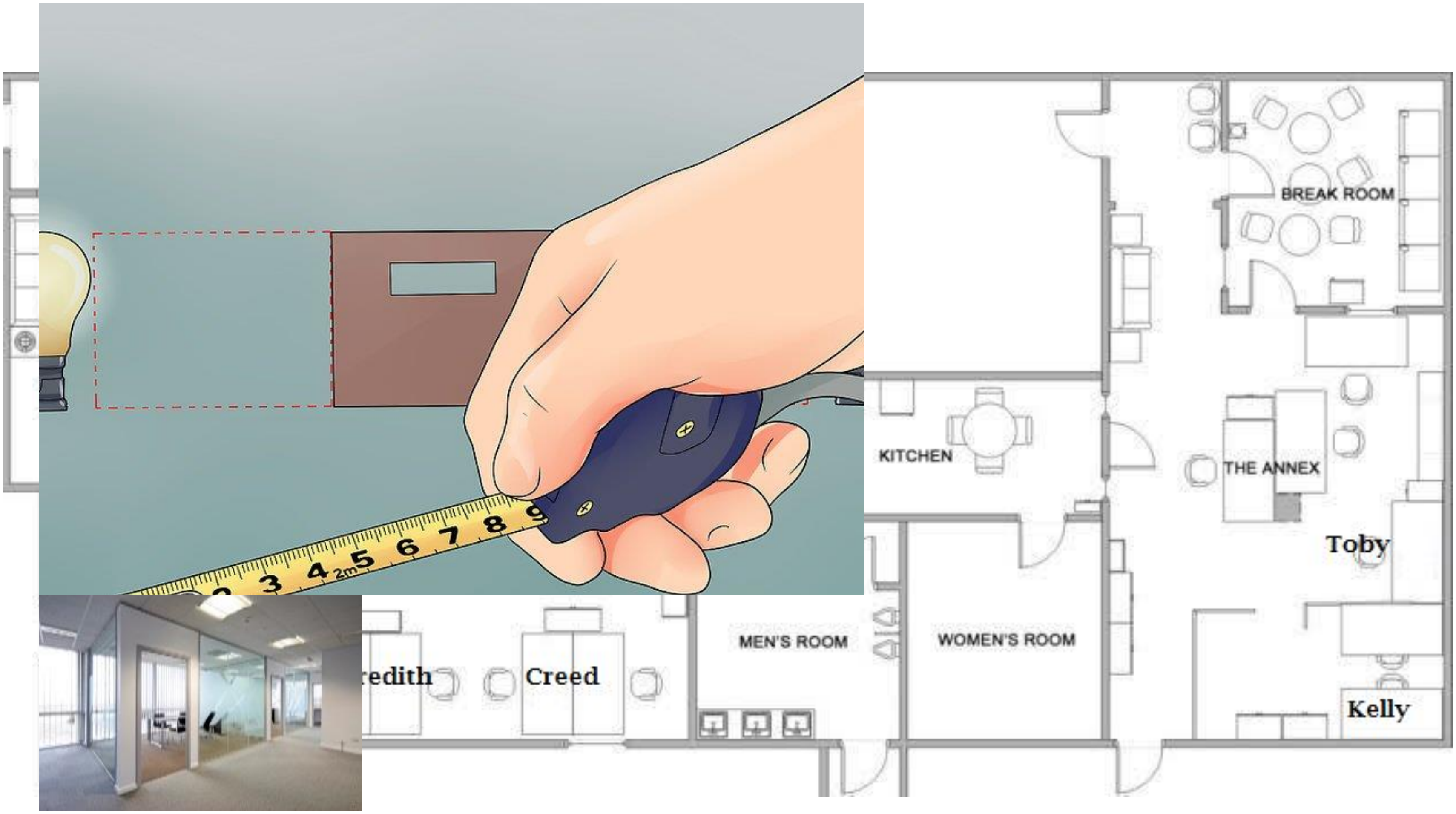
- ❑ Adjust luminaires, ISO standards, predefined scenarios, etc…



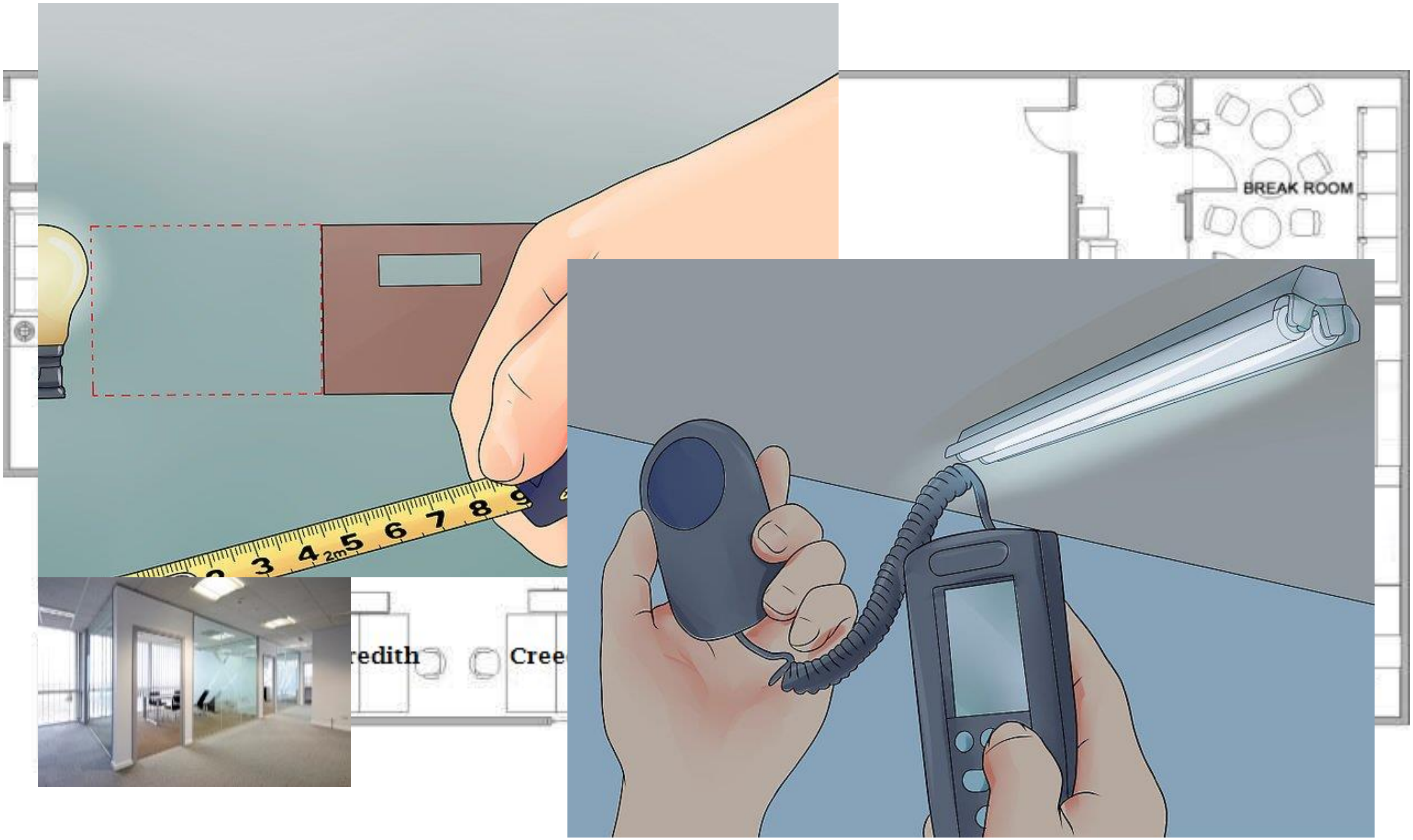
Light systems commissioning



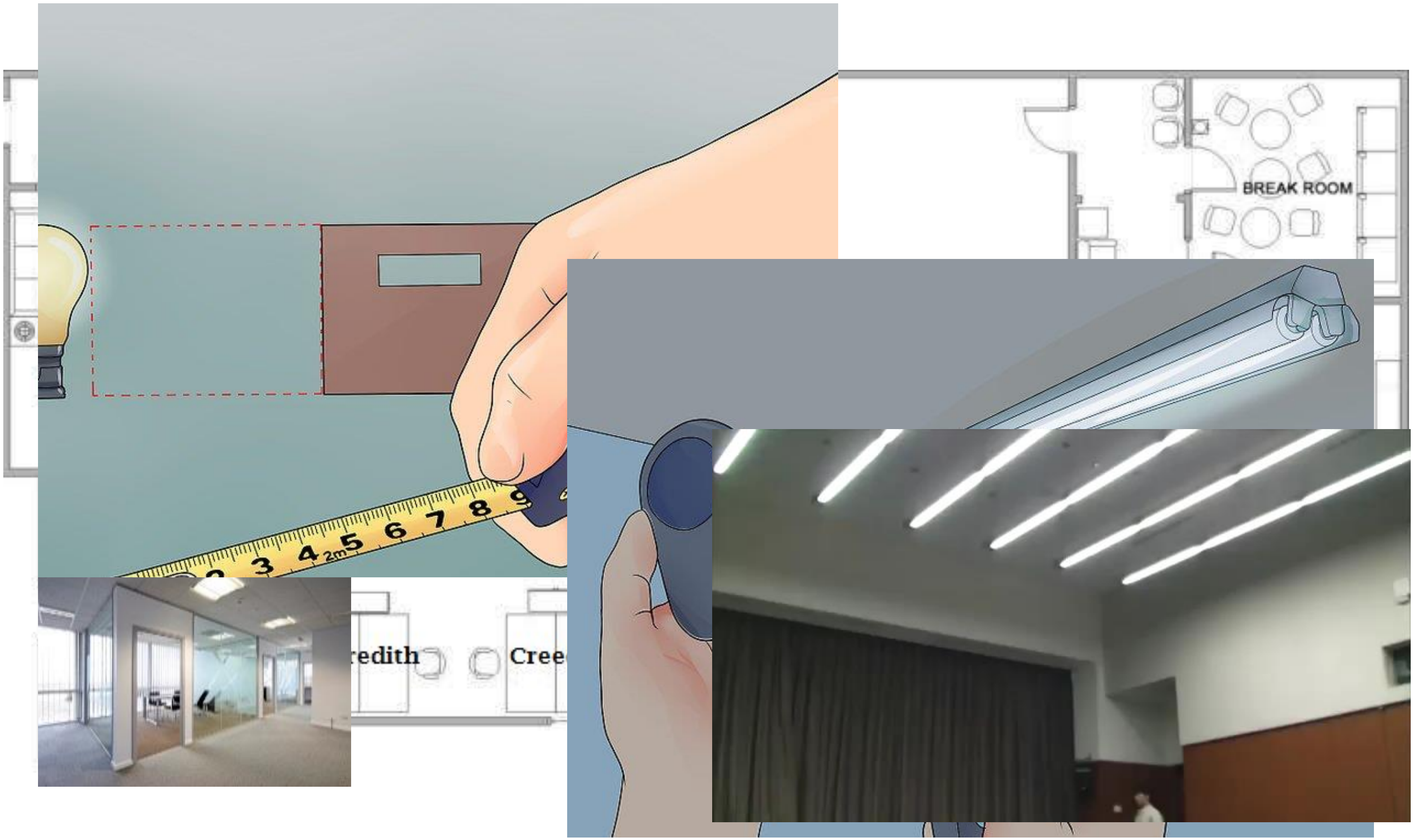
Light systems commissioning



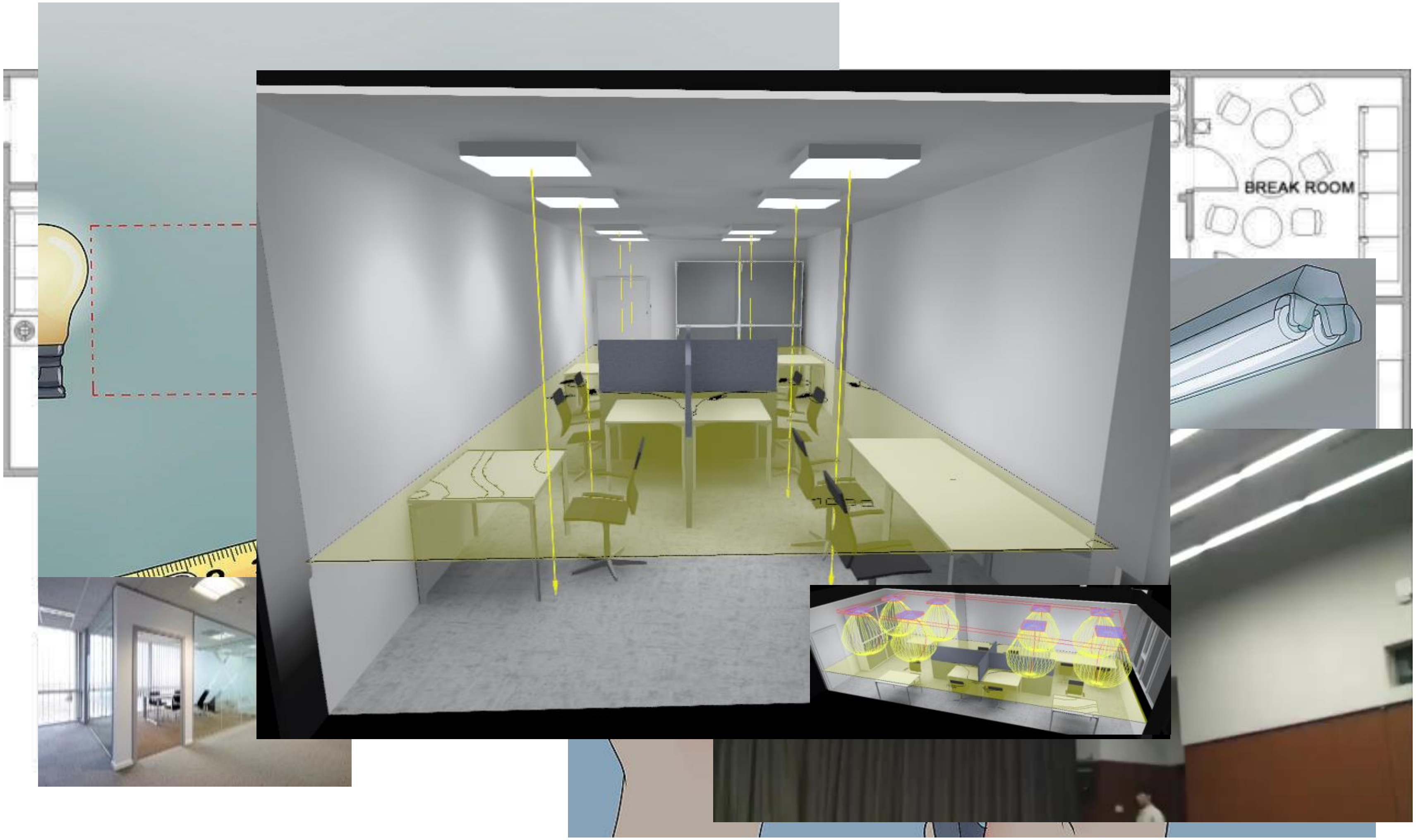
Light systems commissioning



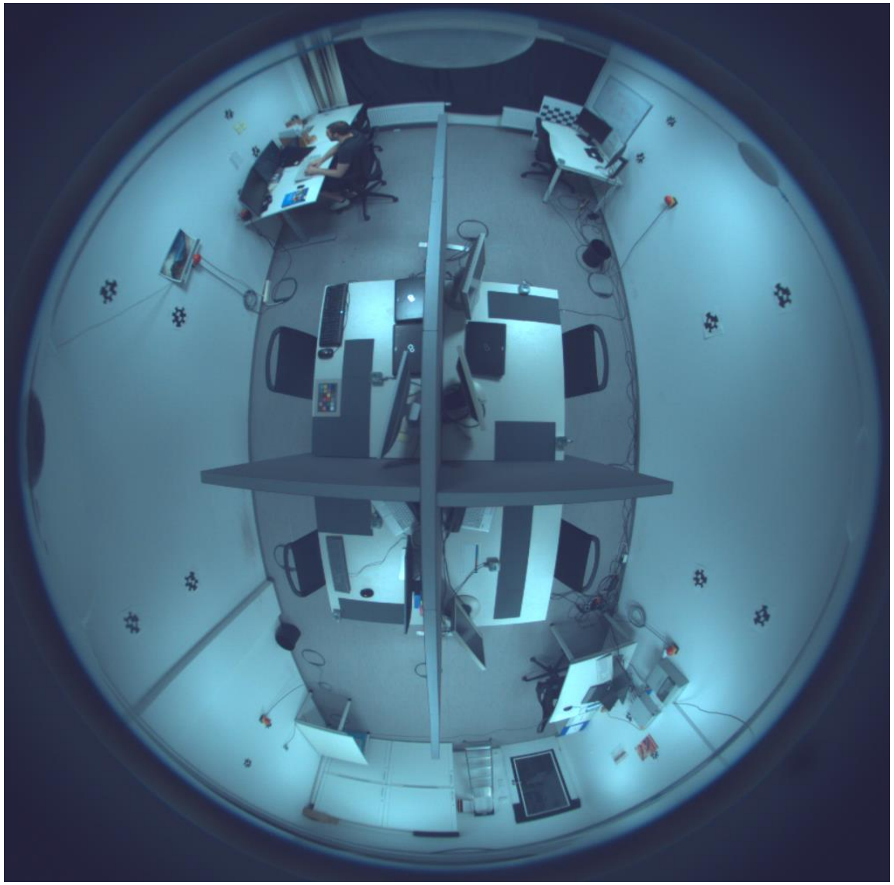
Light systems commissioning



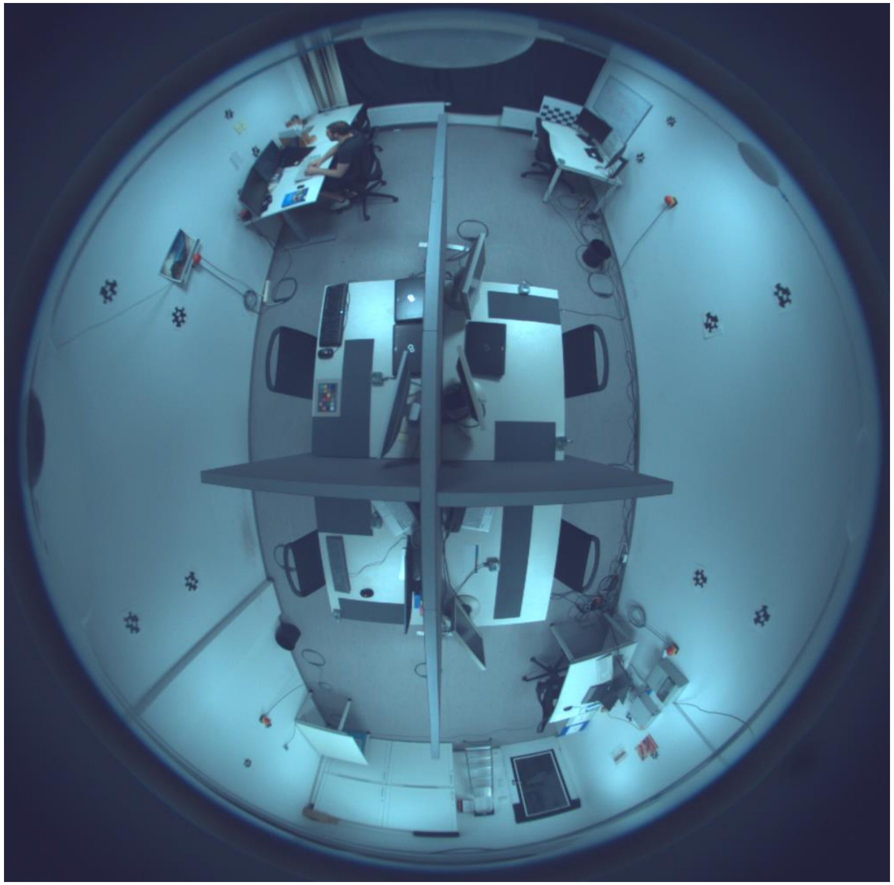
Light systems commissioning

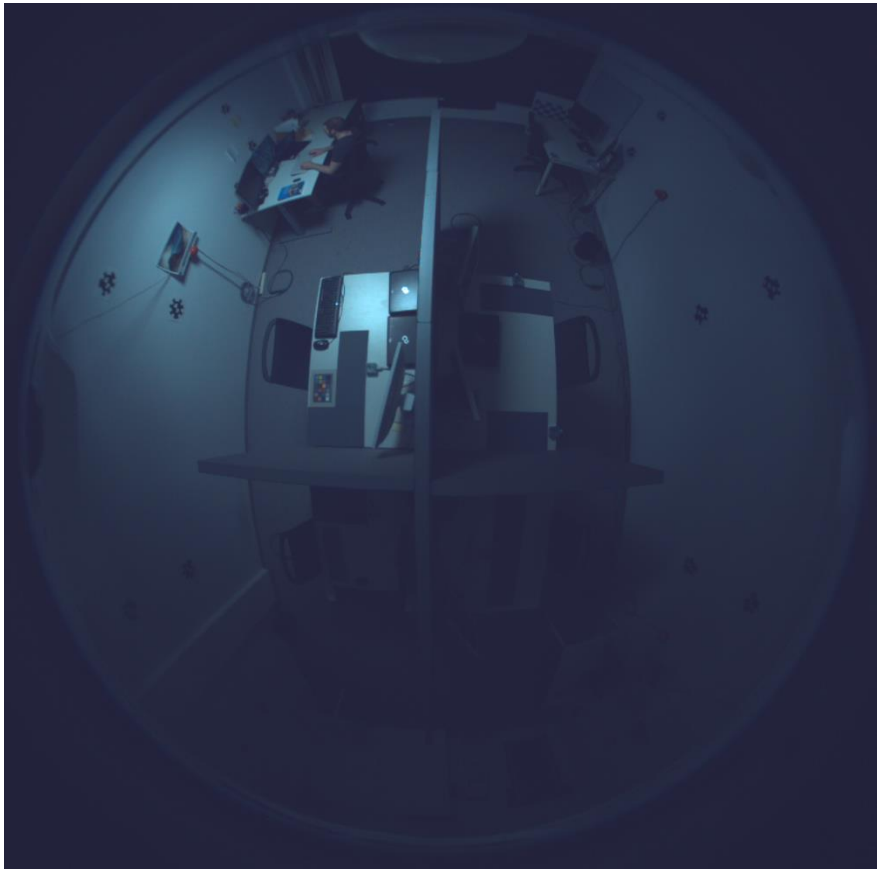


“Invisible Light Switch”

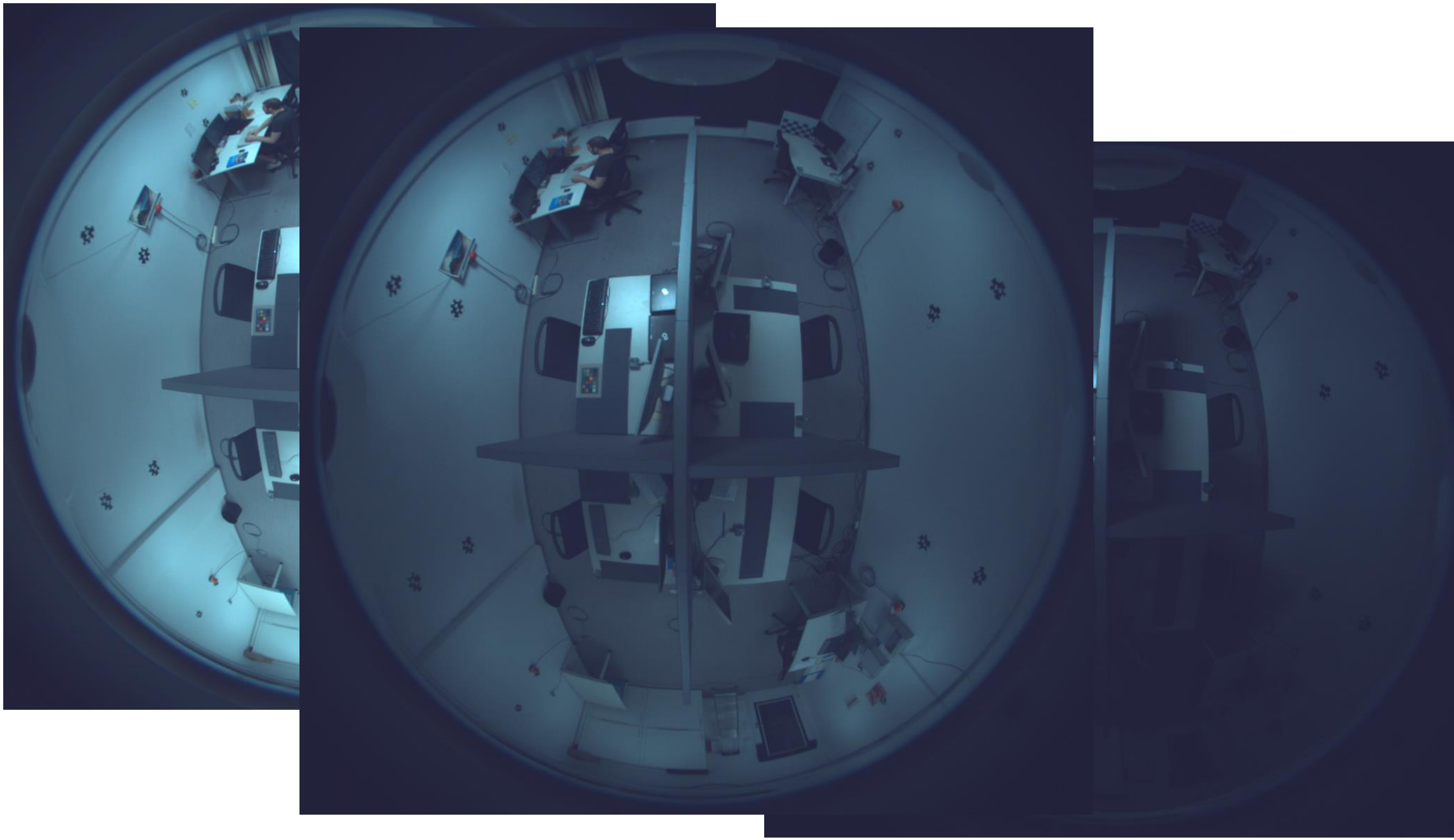


“Invisible Light Switch”





“Invisible Light Switch”



Towards a camera-aided light modeling system

Light in the lighting field

Measuring light…

Light & scene modeling

Radiosity

Practical examples

Applications

Conclusion & future work

Conclusion

Summary

- ❑Light from lighting field
- ❑First light simulation solution with RGBD input
- ❑Radiosity model for real life scenarios
- ❑Meaningful sufficient and reliable such a solution could be
- ❑Remarkable results compared to simulation software

Future work

- ❑On the fly albedo estimation
- ❑More complicated material properties, more complex BRDF
- ❑Dynamic scenes and natural light
- ❑Limitations of partial geometry
- ❑Full automatic solution, estimating light sources positioning and intensity

References

K. Krsl, C. Luksch, M. Schwrzler, and M. Wimmer. LiteMaker: Interactive Luminaire Development using Progressive Photon Tracing and Multi-Resolution Upsampling. In M. Hullin, R. Klein, T. Schultz, and A. Yao, editors, Vision, Modeling & Visualization. The Eurographics Association, 2017.

B. Beckers and P. Beckers. Fast and accurate view factor generation. FICUP, An International Conference on Urban Physics, 09 2016.

L. Masset, O. Bruls, and G. Kerschen. Partition of the circle in cells of equal area and shape. Technical report, Structural Dynamics Research Group, Aerospace and Mechanical Engineering Department, University of Liege, ‘Institut de Mecanique et G enie Civil (B52/3), 2011.

Relux Informatik AG: ReluxDesktop.

MIT EECS 6.837, Durand and Cutler, computer graphics course presentation on radiosity and advanced global illumination.

Theodore Tsesmelis, Irtiza Hasan, Marco Cristani, Alessio del Bue, Fabio Galasso, RGBD2lux: Dense light intensity estimation with an RGBD sensor, IEEE Winter Conference on Applications of Computer Vision, 2019.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Sklodowska-Curie grant agreement No 676455.

Thank you for your attention!

Questions?