



# Wireless Multimedia Sensor Networks

## A brief overview and some challenges

**A. Mostefaoui**

Franche Comte Computing Lab.(LIFC)

Contact: [Ahmed.Mostefaoui@univ-fcomte.fr](mailto:Ahmed.Mostefaoui@univ-fcomte.fr)

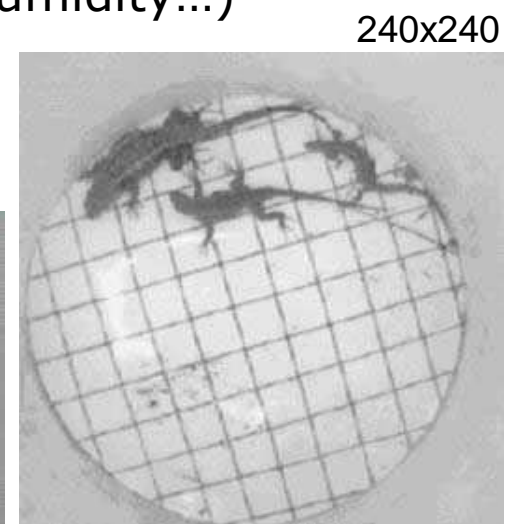
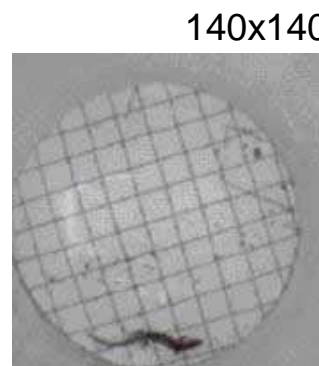
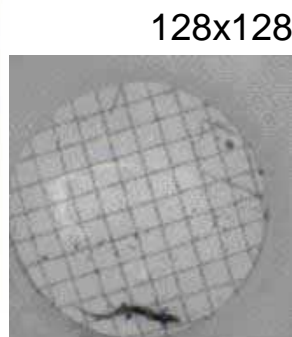
## Outline

1. Introduction : Multimedia Sensor Networks : new challenges or reemerging roles?
2. Data adaptation vs energy consumption : prolonging the network lifetime.
3. data fusion for surveillance purposes.
4. System architecture.
5. Conclusion and future works.

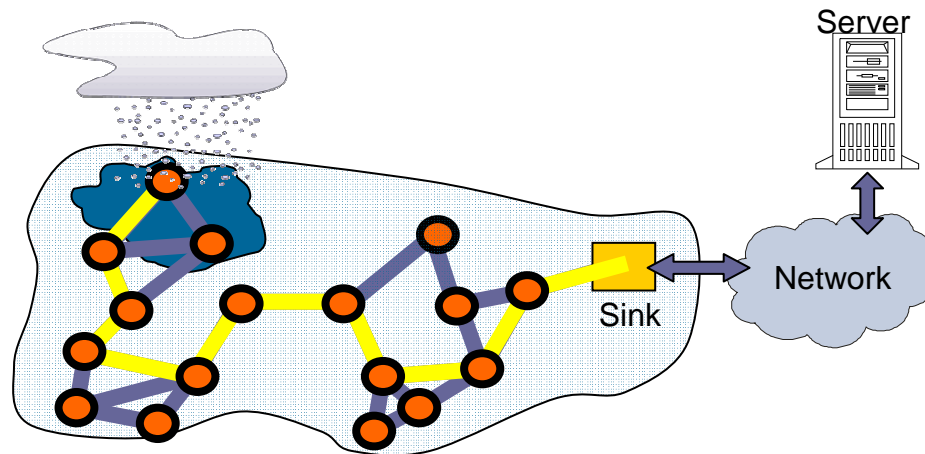
- Multimedia Sensor Networks ?
  - Set of sensors with the ability to collect multimedia data :
    - video, audio, image,
    - scalar data (temperature, humidity...)



Cyclops video board on Mica notes



- Multimedia Sensor Networks ?
  - Communicate through a wireless network



- Characteristics:
  - High data rate (Mbps)
  - Real time requirements (processing and transmission)
  - QoS issues
  - Rich media (interpretation issues)

- Why now ?
  - Availability of cheap multimedia hardware
    - e.g., CMOS camera



© STMicroelectronics

- Advances in hardware: miniaturization of components (chips, embedded systems, radiofrequency communication...)

- Applications:
  - Large scale video surveillance system
  - Health care applications
  - Home automation applications
  - Etc.



## Challenges

*"In addition to the ability to retrieve multimedia data, Wireless Multimedia Sensor Networks will also be able to **store, process in real-time, correlate** and **fuse** multimedia data originated from heterogeneous sources"*

A.F. Akyildiz et al.\*



## Challenges (wsn vs wmsn)

- Coverage issues
  - WSN: sensing range is usually considered as a circle
  - WMSN:
    - we deal with “views” (cone) with zooming capabilities and rotation “orientation” possibilities.
    - Better information could be captured from far multimedia sensors (global understanding of the observed scene)

## Challenges (wsn vs wmsn)

- Fusion issues
  - WSN: averaging scalar sensor measurements.
  - WMSN:
    - Merging two images or videos does not make sense!
    - We are more concerned with content merging (observed scene)
    - How to extract data content from images and videos?

## Challenges (wsn vs wmsn)

- QoS issues
  - WSN: data uncertainty (corrupted measurements due to environmental conditions for instance)
  - WMSN:
    - Data acquisition: high resolution does not necessary lead to a better understanding of the data content!
    - ...

## Challenges (wsn vs wmsn)

- QoS issues
  - WMSN:
    - Example :



333x358 16M colors, no light



167x180 16 colors, light



167x180 BW (2 colors), light

## Challenges (wsn vs wmsn)

- QoS issues
  - WMSN:
    - Real-time delivery:
      - avoidance of jitter
      - Critical event detection

- 320x200
- 30 fps
- 256 gray scale
- 15Mbps raw



- 320x200
- 2 fps
- 4 gray scale
- 256 Kbps raw

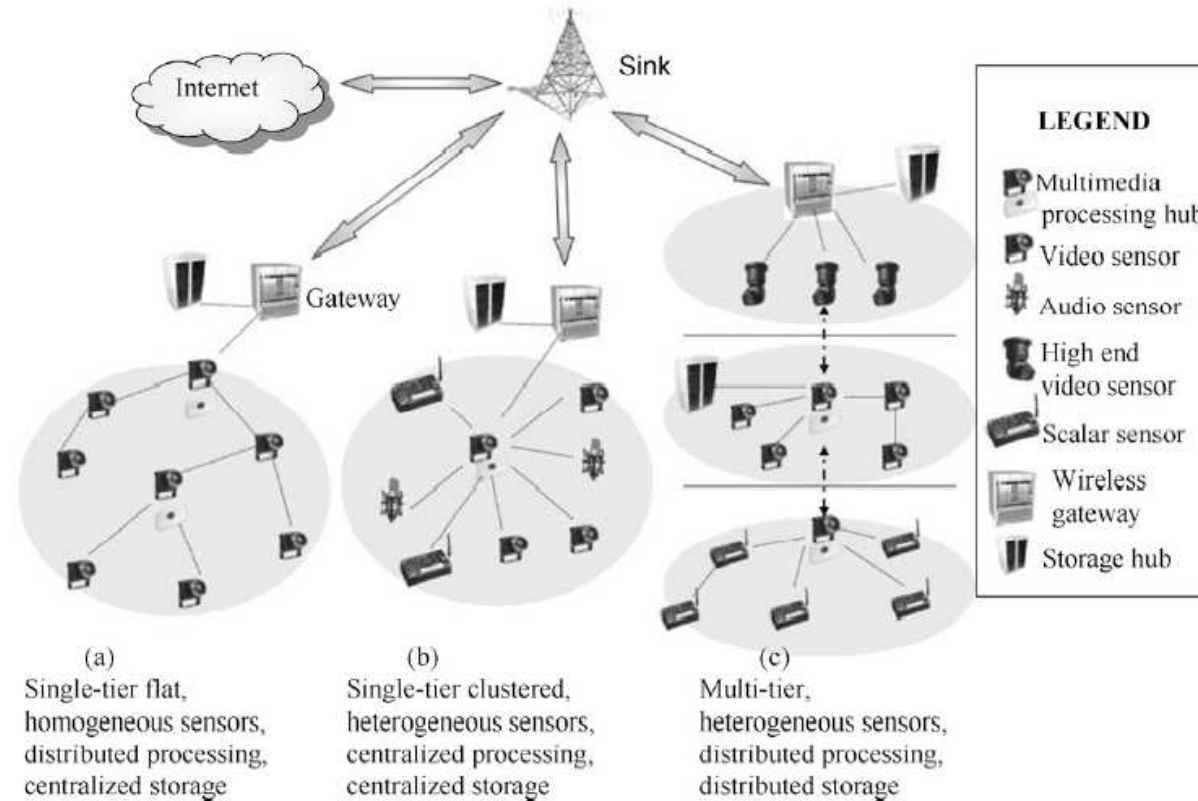
- End-user real-time processing: in order to extract content features for instance.

## Challenges (wsn vs wmsn)

- Heterogeneity issues
  - WSN: scalar heterogeneity (humidity, pressure, temperature, etc.)
  - WMSN:
    - Scalar sensors,
    - Multimedia sensors (video, audio),
    - Intermediate storage/processing hubs,
    - Etc.

# Challenges (wsn vs wmsn)

- Heterogeneity issues
  - WMSN: reference architecture



## Challenges (wsn vs wmsn)

- Energy optimization issues
  - WSN: energy consumption is the main optimization concern.
  - WMSN :
    - Also true...
    - But: more depended of the target application (video surveillance applications, ...)
    - Cross-layer optimization including **acquisition, coding, transmission.**





# Data adaptation vs energy consumption

## The problem

- In wireless Ad Hoc environments:
  - In addition to their own workload, nodes have to perform network functionalities (routing and relaying)
  - Nodes are energy constrained.
  - Each node, participating in the wireless ad hoc network, wants to prolong its own lifetime by reducing its “**relaying tasks**”
  - If each node acts as above, the network will broke down!

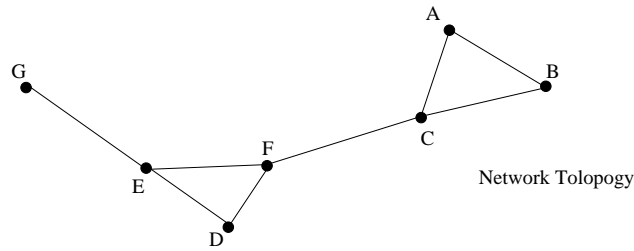


L I F C

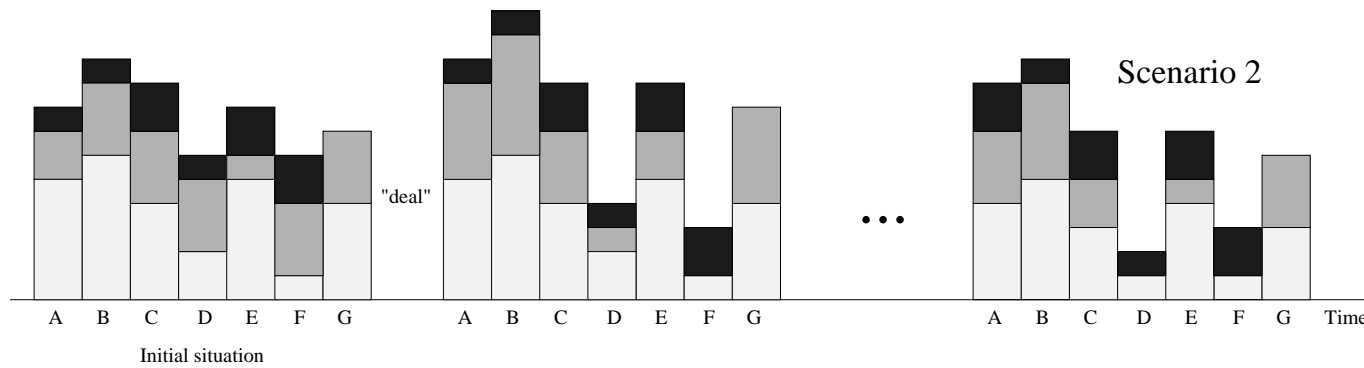
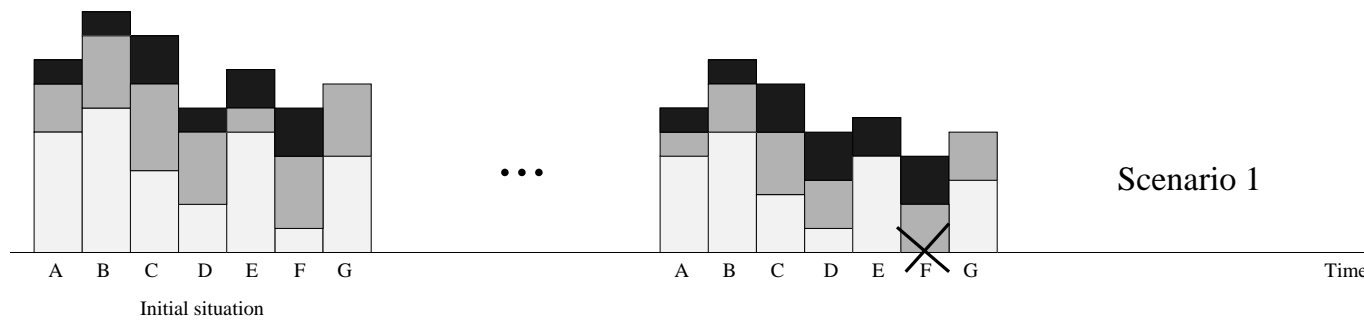
UNIVERSITÉ DE FRANCHE-COMTÉ



# Motivation



- Energy Provision
- Application workload
- Network workload (relaying)



## Objective

meets the node interest with the network one.  
We are looking for a “deal” (*an approach*)  
such that:

- Each node, participating in the network, has the guaranty that its participation is conditioned by its own energy provision.
- Ensure a long (max) lifetime to the network.
- Ensure fairness to all nodes of the network



## Our approach

- A totally distributed algorithm (i.e., each node is running this algorithm).
- Nodes communicate only with their neighbors.
- Analytically proven to ensure fairness to all nodes.

## Notation

Let us use the following notation:

- $E_i^{(k)}$ : energy provision of node  $i$  at time slot  $(k)$ .
- $T_{i,j}^{(k)}$ : number of class  $j$  tasks that node  $i$  holds at time slot  $(k)$ .
- $e_{i,j}$ : energy spent by node  $i$  to execute one task from class  $j$ .
- $N_{i,j}^{(k)}$ : number of class  $j$  tasks executed by node  $i$  during time slot  $(k)$ .

# Notation

–  $k > S$

Node 1 ratio  
value at instant  
(k)

Energy needed for  
node 1 to execute a  
task from class J

Remaining tasks of  
class J at instant (k)

$$\frac{\sum_{j=1}^m T_{1,j}^{(k)} e_{1,j}}{E_1^{(k)}} = \dots = \frac{\sum_{j=1}^m T_{N,j}^{(k)} e_{N,j}}{E_N^{(k)}}$$

Remaining energy  
provision at instant  
(k) for node 1

Let us use the following notation:

Node i ratio value at instant k

Remaining tasks of class J at instant k

Energy needed for node i to execute a task from class J

$$x_i^k = \frac{\sum_{j=1}^m T_{i,j}^{(k)} e_{i,j}}{E_i^k}$$

Remaining energy provision at instant k for node i



## Energy Aware Assignment Algorithm

```

if  $V_i^k \neq \emptyset$  do
    1. Exchange  $x_i^{(k)}$  with each  $v_n \in V_i^{(k)}$ 
    2. Compute  $x_i^{(k+1)} = x_i^{(k)} + \sum_{n \in V_i^{(k)}} A_{iv}^{(k)} (x_v^{(k)} - x_i^{(k)})$ 
    if  $(x_i^{(k+1)} - x_v^{(k)} > \epsilon)$  do
        if  $(x_v^{(k)} - x_i^{(k)} < 0)$  do
            3. Compute  $\alpha_{i,v}^{(k)} = A_{v,i}^{(k)} \left( T_i^{(k)} - \frac{E_i^{(k)} x_v^{(k)}}{e_{i,\theta}} \right)$ 
            4. Send  $|\alpha_{i,v}^{(k)}|$  tasks of class  $\theta$  to node v
        endif
    endif
endif
endif
endif
    
```



## Energy Aware Assignment Algorithm

- Diffusion matrix:
  - We used a symmetric doubly stochastic matrix
- We proved that the above algorithm converges after a certain number of iterations.

## Experimental settings

### Energy consumption:

- Data processing:
  - Four types of tasks: *baseline* (system tasks), *processing-intensive* (FFT), *storage intensive* and *communication-intensive*.
  - Two types of computer (*Dell Latitude C600* and *Crossbow Stargate platform*)
- Data transmission:
  - *Lucent WaveLAN IEEE 802.11 Silver* card.

## Experimental settings

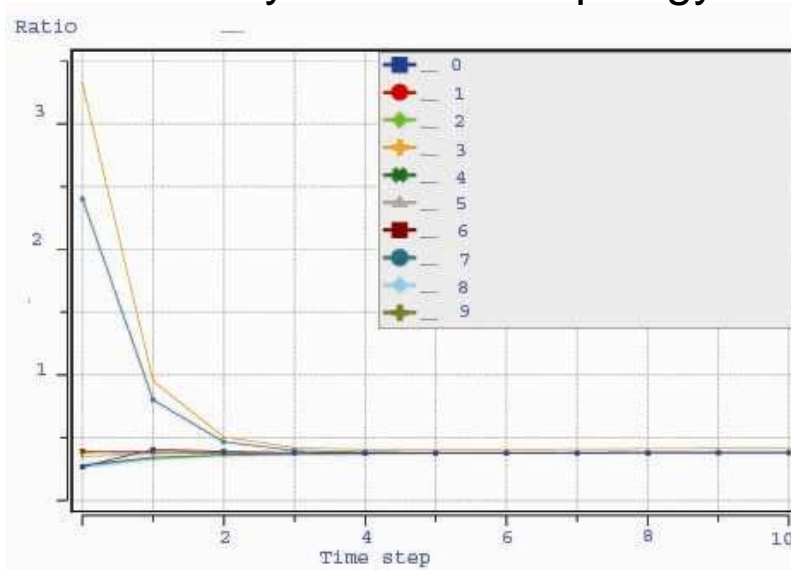
### Settings :

- **Types of nodes:** two types
  - PC like nodes (*Dell Latitude C600*)
  - PDA like nodes (*Crossbow Stargate Platform*)
  - 5~20 nodes in the network.
- **Network topology:** three topologies
  - linear topology,
  - fully-connected topology
  - mixed topology:
- **Initial energy provision:**
  - Varied in order to get a variation in the *standard deviation* between nodes.

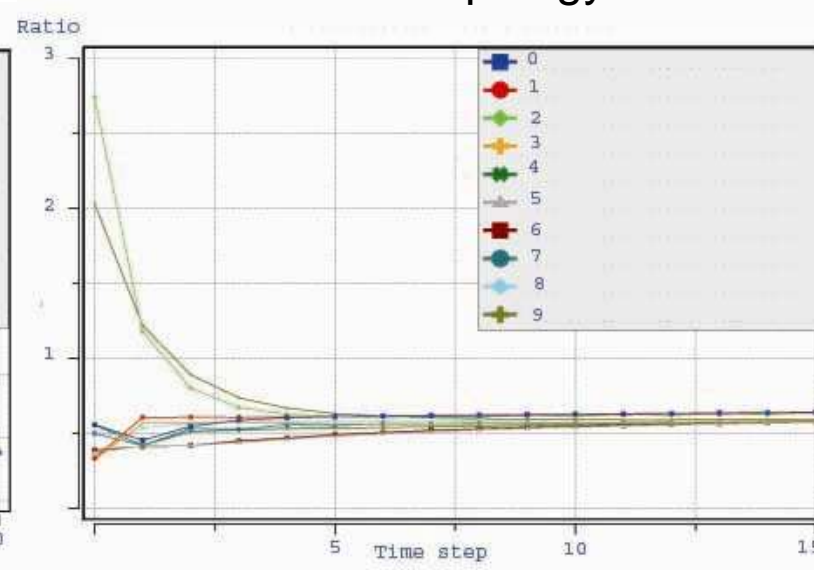
# Data adaptation vs energy consumption

## Convergence (equity)

Totally connected topology



Linear topology



## Preliminary results

Results:

Fully connected topology	Linear topology	Mixed topology
55.5%	49.01%	50%

Network lifetime improvement



# Continuous 3D representation for multimedia sensor networks

## The challenge

### Traditional surveillance infrastructure



What happen when we are dealing with dozens or hundreds of video sensors ?



## Our approach

- Targeted to the video surveillance system
- 3D real-time reconstruction : the views of the sensor nodes are merged in the 3D scene of the observed region
- Allows a comprehensive representation of the scene
- Better supports spatio-temporal requests :
  - “Where is the busiest place in the building?”
  - “Where is the place where people spend most of their time?”

## The Problems

- Deporting image processing to sensors
  - Reduce the network load by sending only some data about detected objects in the image
- Fusion of these data
- 3D representation of the resulted information for the human observer

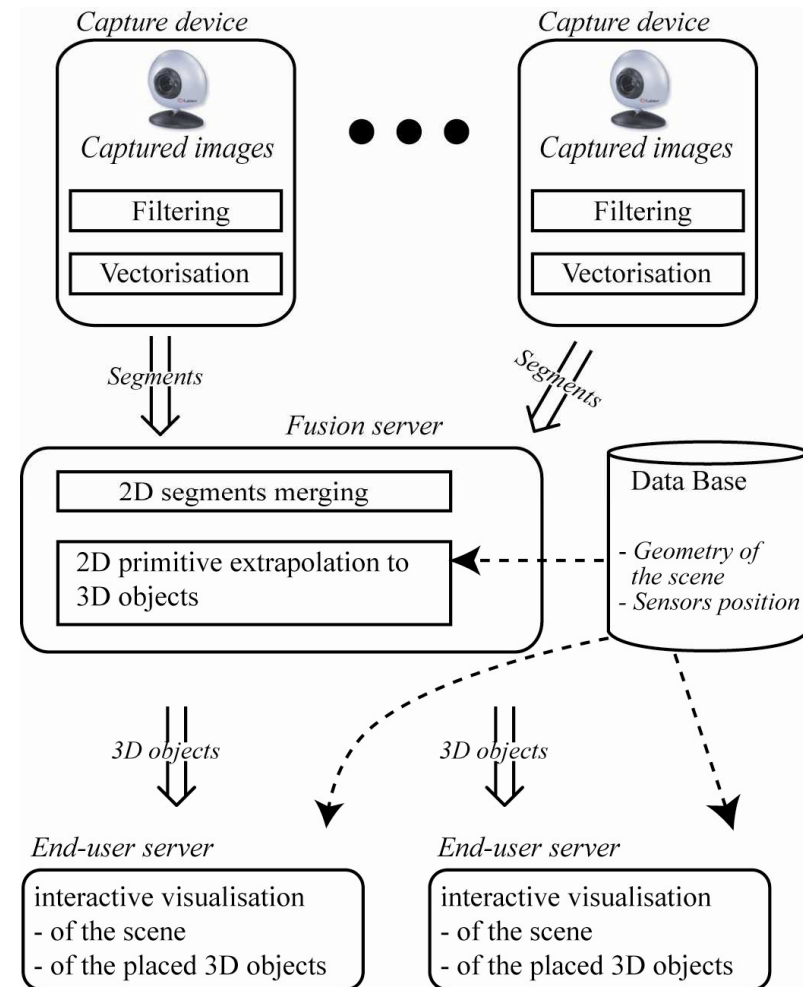


L I F C

UNIVERSITÉ DE FRANCHE-COMTÉ

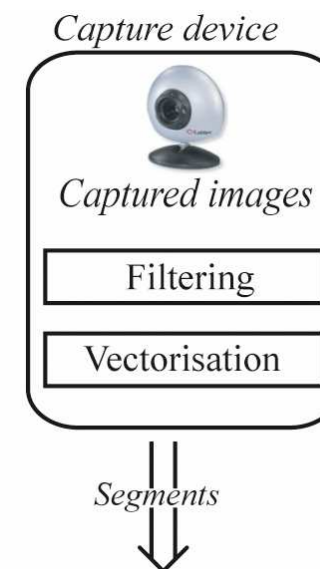
## System architecture

- Three main components:
  - The Capture device
  - The fusion server
  - The end-user server



## System architecture

- The capture device
  - Capturing and processing the video
    - Hardware architecture (32 bits processors)
    - processing (simple testing methods)
      - Difference between images
      - Borders extraction
      - Vectorisation
  - Reduces the amount of data to send to the fusion server
    - Set of segment coordinates.

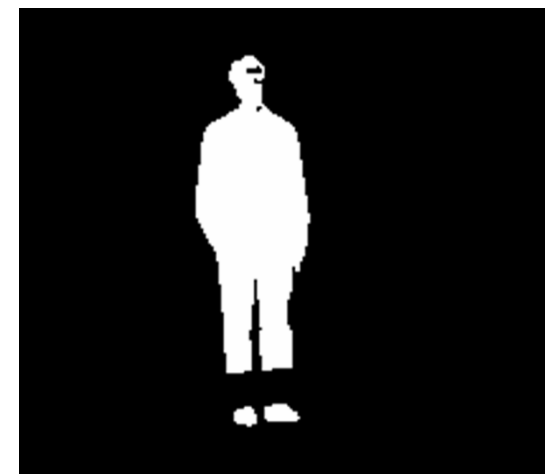


## System architecture

- Image capture
- Difference with the reference image

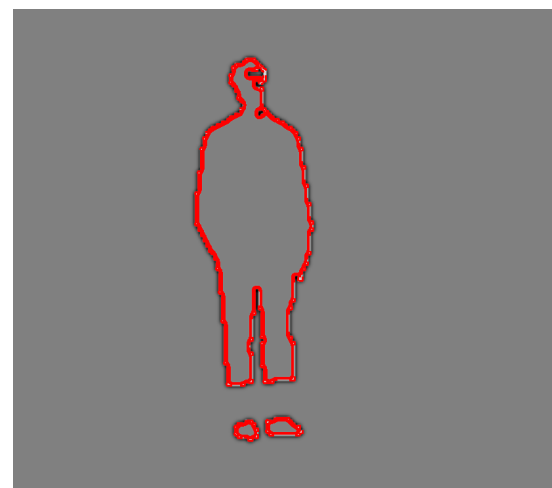
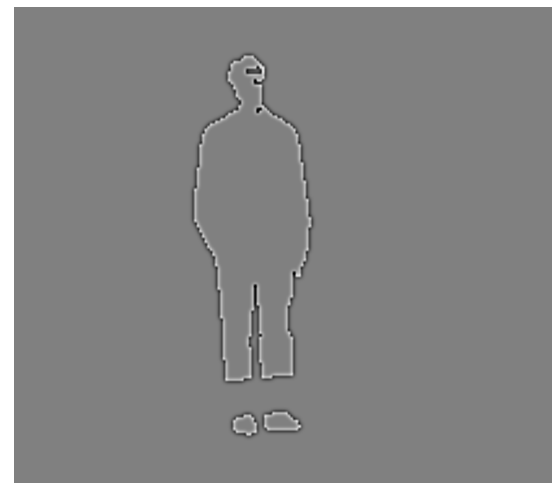


- Thresholding to obtain black and white image



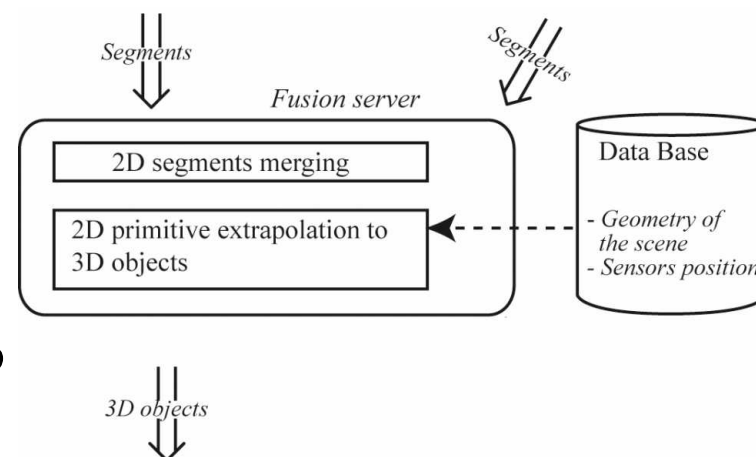
## System architecture

- Filtering to extract borders.
- Vectorisation to obtain a set of segment coordinates.
- Sending the coordinates to the fusion server

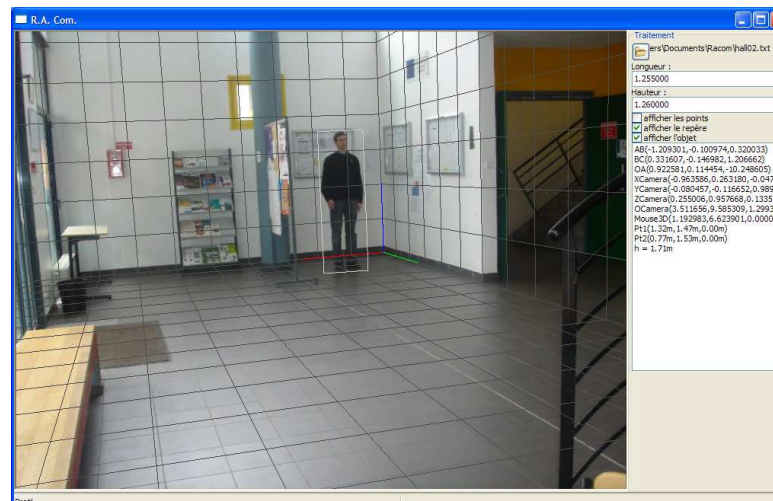
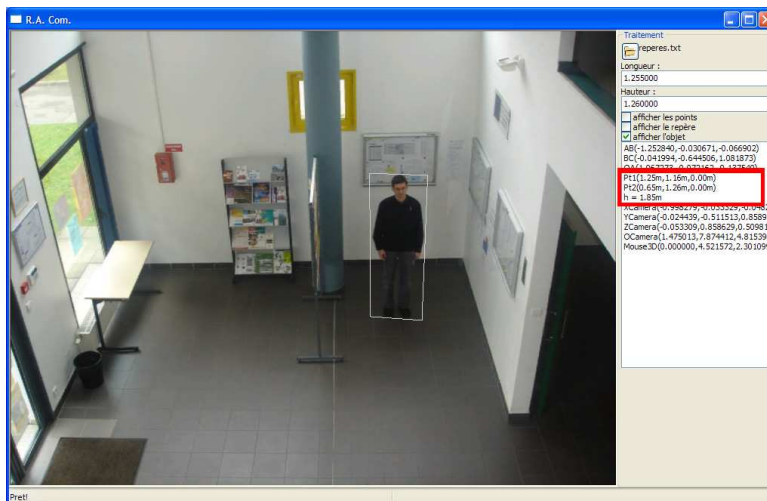


## System architecture

- The fusion server
  - Merges the data sent by all the capture devices
  - Places the detected objects into the 3D environment
  - Use a 3D database
    - 3D Model of the scene
    - Position and direction of each camera
      - Needs a calibration step



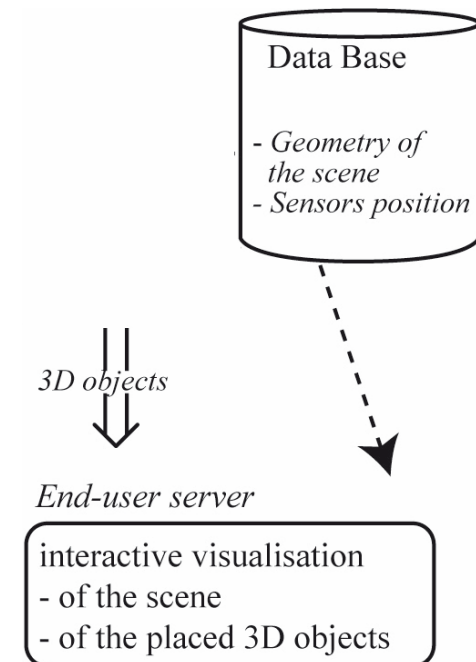
- 3D interpretation:
  - Using simple 3D hypothesis
    - People standing on the floor
  - 2D data are placed in the 3D environment
    - Possibility of merging data (average position....)





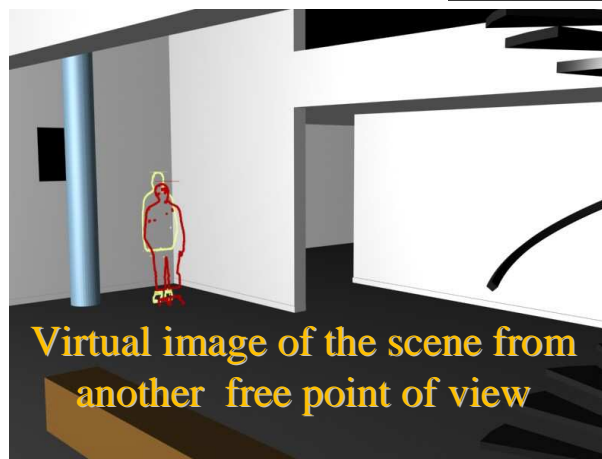
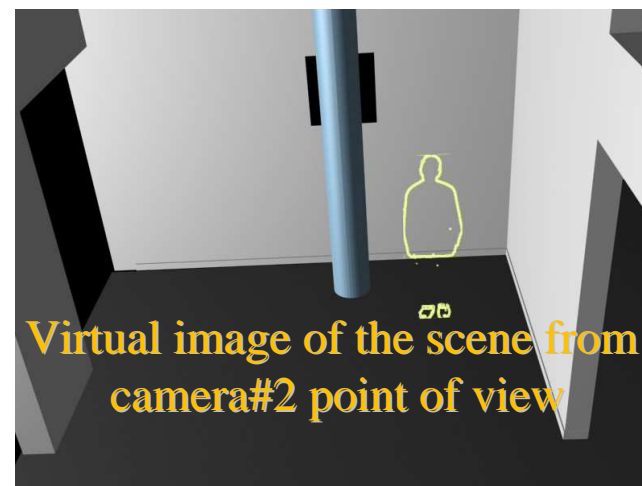
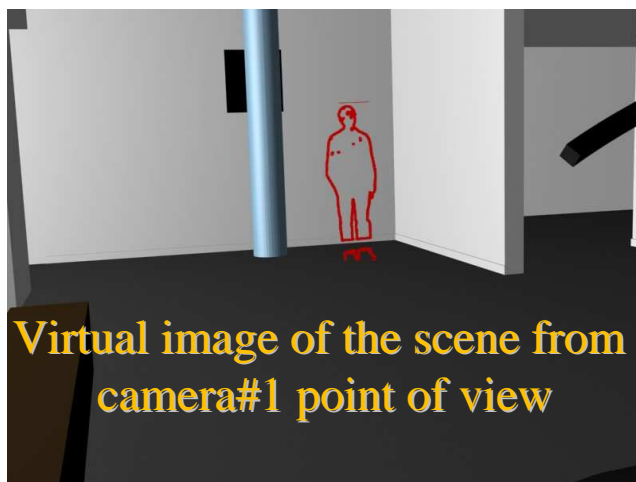
# System architecture

- The end-user server:
  - Interface representing in 3D
    - The geometrical model of the scene
    - The detected objects
      - Viewed from camera positions
      - Calculated from all other positions in the building
  - Using Virtual Reality software
    - Virtools (Dassault system)



# System architecture

- The end-user server:



## Preliminary Results

- Capture device image processing capabilities
  - 27 frames/second for 320x200 video images
  - 2 frames/second for 640x480 video images

## Conclusion and future work

- Key points:
  - Data processing located on the sensors allows a significant network traffic reduction
  - 3D real-time reconstruction allows:
    - Comprehensive understanding of the observed scenes
    - Complex spatio-temporal requests
    - Advanced user interaction
- Limitations:
  - Not always relevant (depends on the target application)



Thank you for your attention!...

Questions are welcomed ;-)