



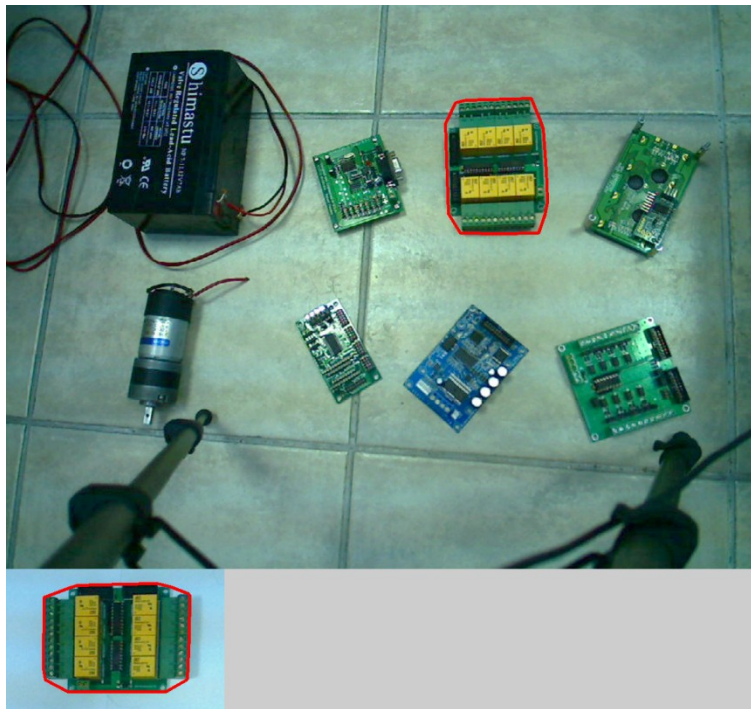
## SentiSight SDK 2.0: Object Recognition.

# SentiSight SDK 2.0 : Object Recognition.

- ▶ What is SentiSight?
- ▶ Properties of the algorithm.
- ▶ Examples of objects.
- ▶ References.

# What Is SentiSight?

1. Can detect **WHETHER** and **WHERE** particular object is presented in a scene.



# What Is SentiSight?

2. Can detect **HOW MANY** particular object instances are presented in a scene.





# What Is SentiSight?

3. Can **COMPARE** two pictures.



# Properties of the algorithm.

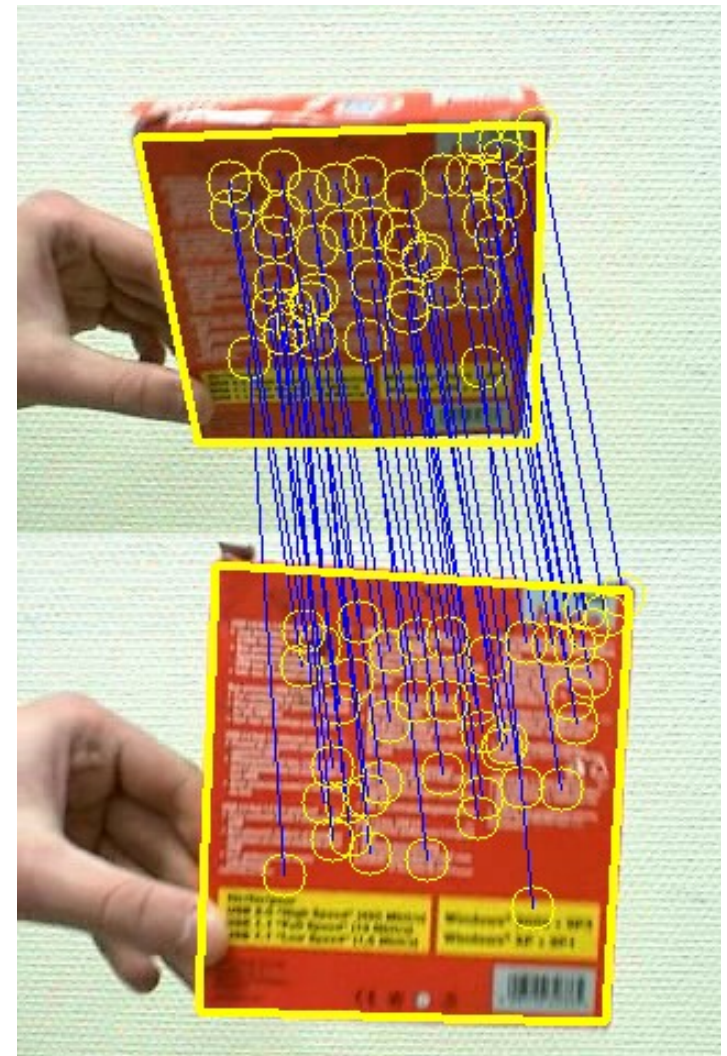
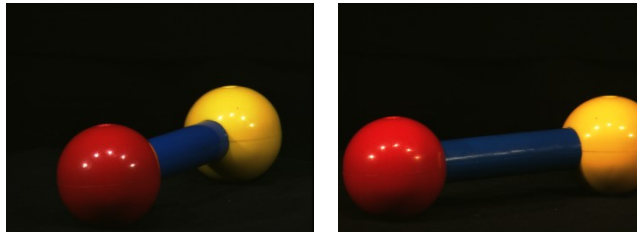
- ▶ SentiSight is very capable!
- ▶ As for every algorithm in the world some conditions should be met in order to get best of the algorithm.

# Properties of the algorithm.

## Local Appearance Based:

In order to be recognized, same object in two different images should have **details (local features)** which **look (appear)** similar.

No common local features...



# Properties of the algorithm.

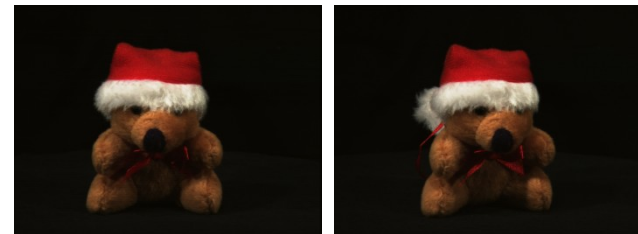
## Lighting conditions:

- ▶ Illumination.
- ▶ Shadows.
- ▶ Reflectance.

Planar objects only have problems with reflectance ...



3D objects have little problems with constant lighting conditions ...



3D objects have problems with varying lighting conditions ...





# Properties of the algorithm.

Transparency:

- ▶ Generally transparent objects are difficult to recognize.

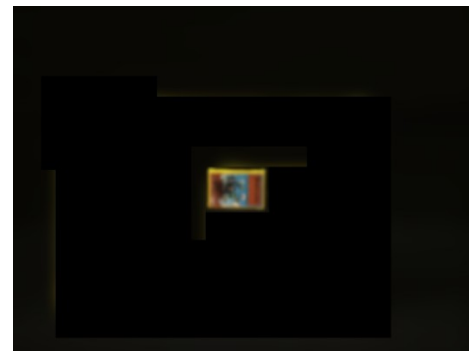


# Properties of the algorithm.

Resolution and Scale (size in image):

- ▶ Objects should contain enough details, and be large enough to be recognized.
- ▶ Scale difference can be up to 2–3 times.

Too big difference ...



# Properties of the algorithm.

## Rotation and Translation:

- ▶ In a plane perpendicular to the camera.
- ▶ The algorithm is generally rotation and translation invariant.



# Properties of the algorithm.

Free rotation:

- ▶ Out of a plane perpendicular to the camera.
- ▶ The algorithm is invariant for rotations up to 10–15 degrees, so add different rotations (views of an object) to the model.





# Properties of the algorithm.

## Rigidity:

- ▶ The algorithm can recognize only rigid objects.
- ▶ At least significant part of the object should be rigid.

At this scale the object has no stable local features and is not rigid ...



# Properties of the algorithm.

## Occlusions:

- ▶ The algorithm is generally robust to occlusions .
- ▶ Occlusions as big as 50%.



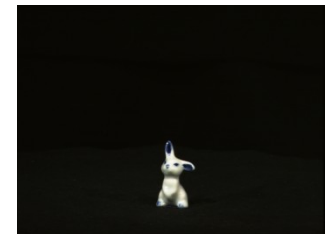
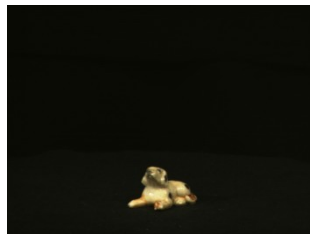
# Properties of the algorithm.

Model formation and recognition:

- ▶ Add different appearances of the object to the model.
- ▶ Model size and recognition time is linear in respect to the number of different appearances.

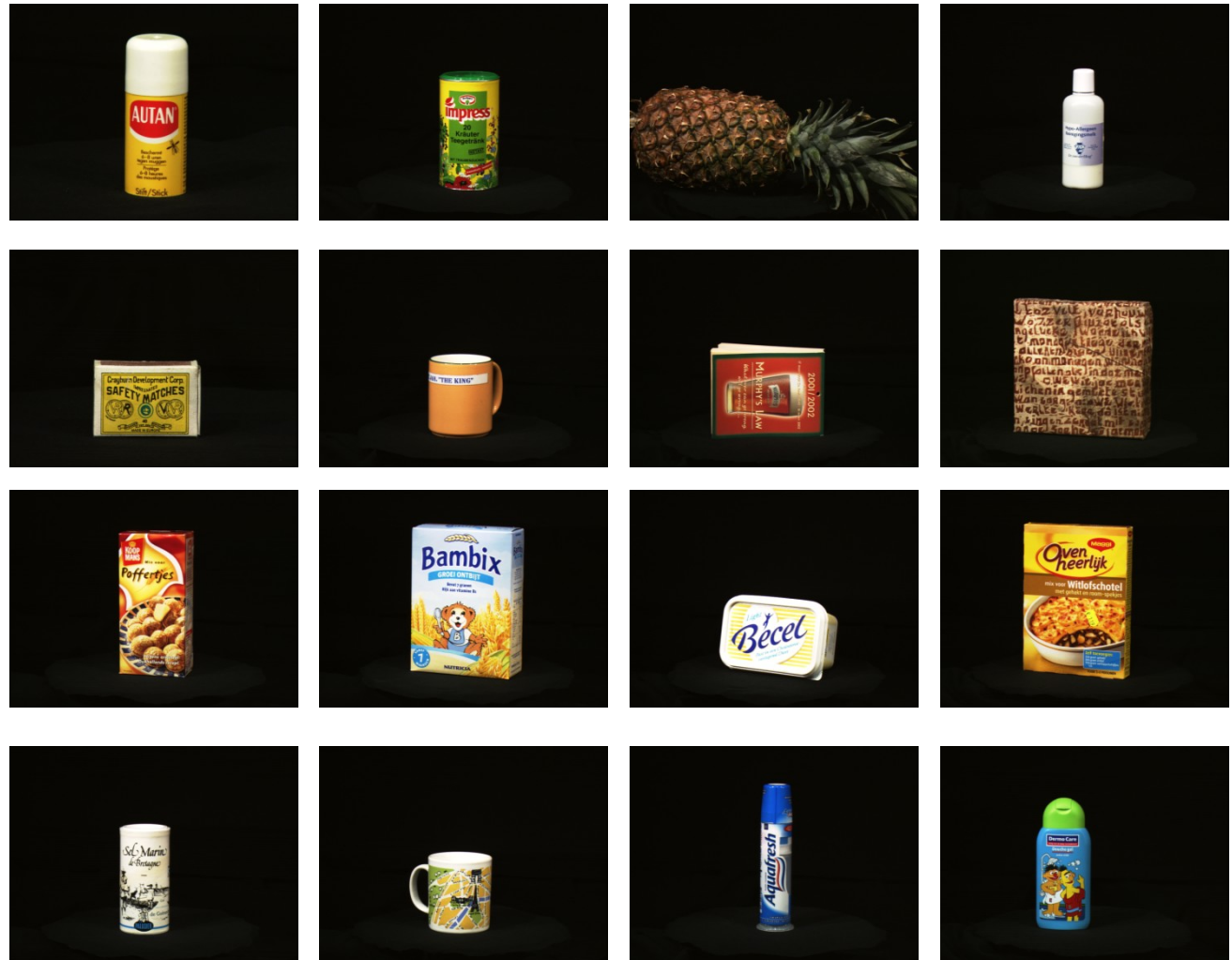


# Examples of recognizable objects ...





# Examples of recognizable objects ...



# Examples of objects which are harder to recognize ...



# References:

Most of the images are taken from the ALOI image database:

<http://staff.science.uva.nl/~aloi/>

# Questions

