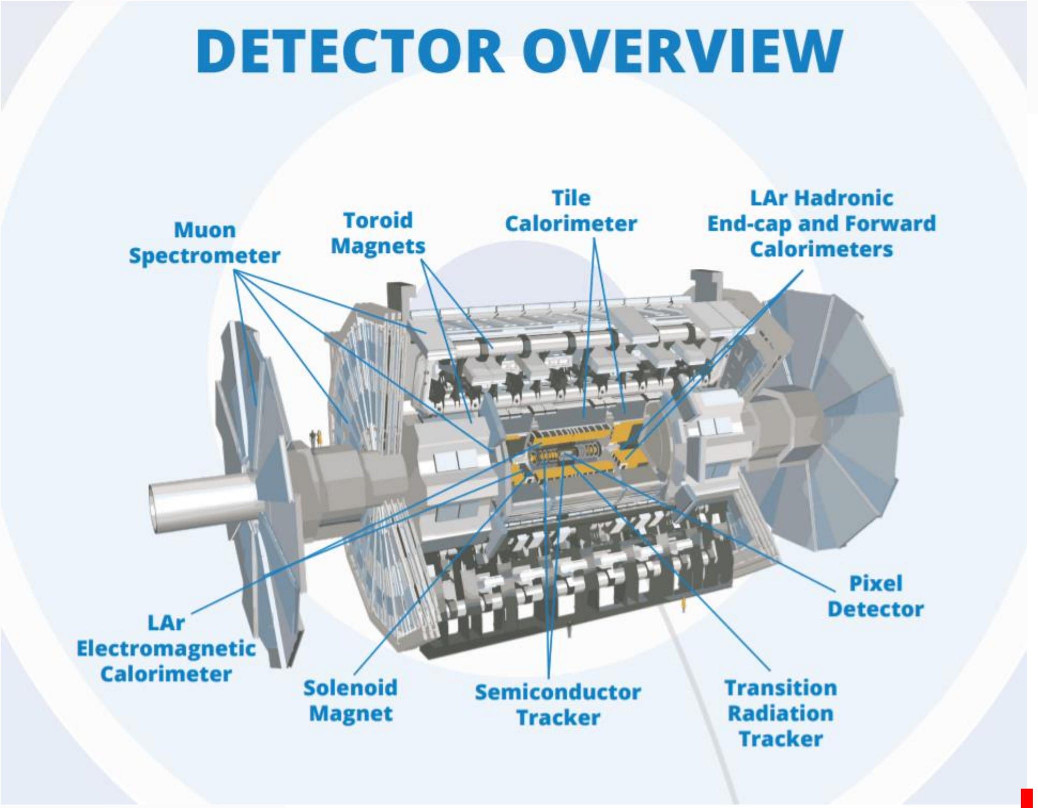
Development of tools for visual inspection of the ATLAS ITk pixel modules : automatic wire bonding quality assessment

Isai Sotarriva
Tokyo Institute of Technology (東京工業大学)

Isai Roberto;Sotarriva Alvarez[A], Atsuhiro;Suzuki [A], Osamu;Jinnouchi[A], Minoru;Hirose[B], Hideyuki;Oide[C] , Koji;Nakamura[C], Manabu;Togawa[C], Kazunori;Hanagaki[C]

[A] Dept. Of Phys., Tokyo institute of Technology, [B] Dept. Of Phys., University of Osaka, [C] The High Energy Accelerator Research Organization (KEK)

Wednesday, September 7, 2022



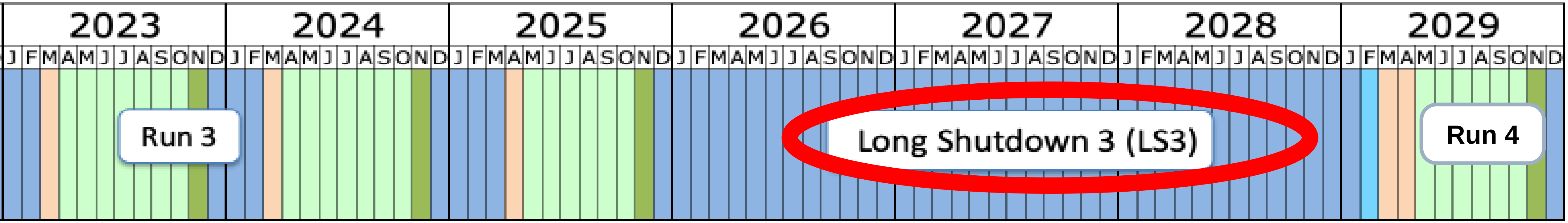
Towards the high luminosity large hadron collider (HL-LHC)

The ATLAS experiment is one of the 2 general-purpose detectors at the LHC.

In order to adapt to the new requirements that come with the increase in luminosity, the inner tracking system (to become the ITk) will be upgraded during the long shutdown 3 (Dec 2025-Jan 2029).

Large hadron collider (LHC)
$L = 1 \times 10^{34} cm^{-2} s^{-1}$

High luminosity LHC (HL-LHC)
$L = 5 \times 10^{34} cm^{-2} s^{-1}$

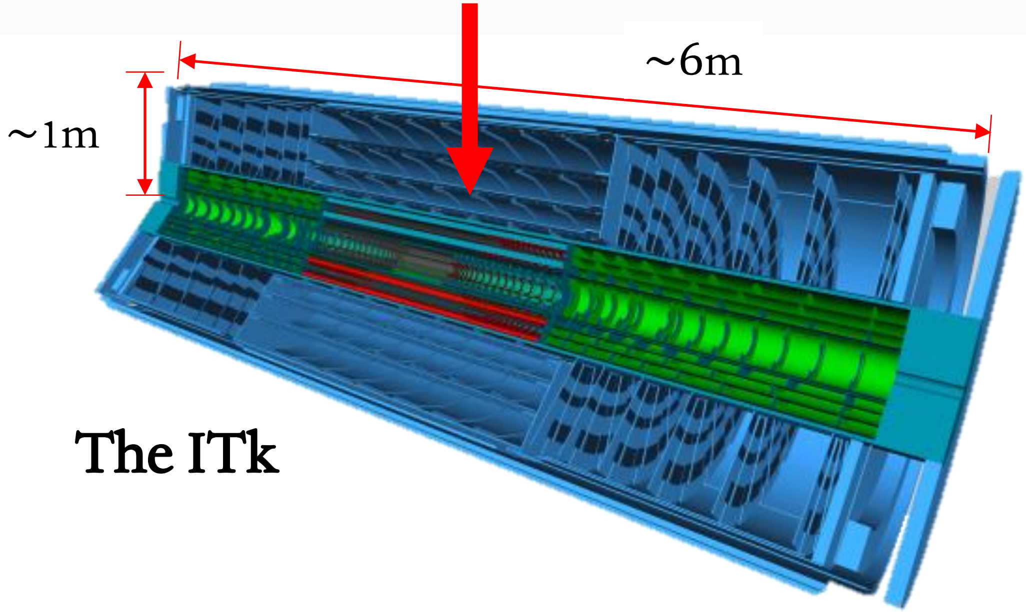


Pixel modules

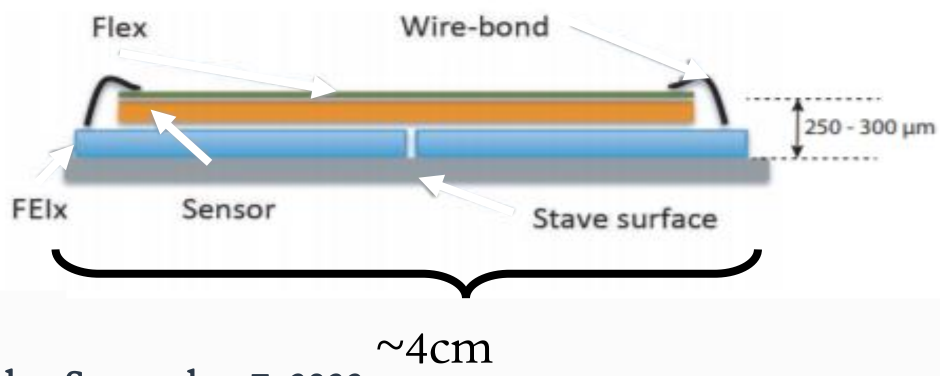
The **ITk** is composed of the pixel and the strip detectors.

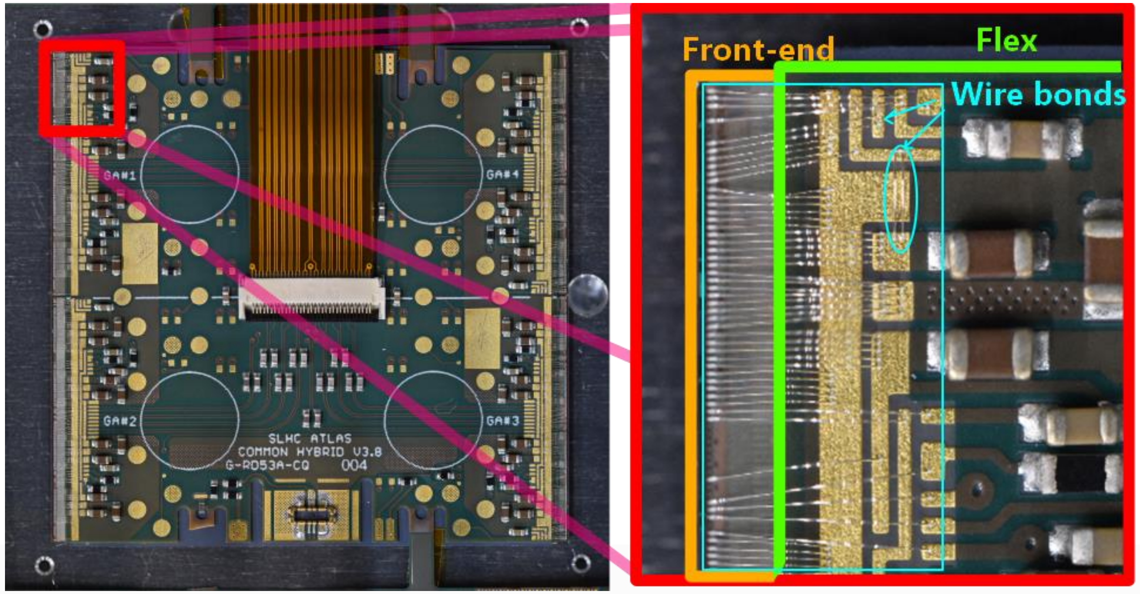
The pixel module consists of 3 main components:
Sensor: Generates an electric charge signal when a charged particle crosses by.
Front end chip: The FE chip integrates the charge generated by the sensor, amplifies and digitizes the signal, and sends the hit information to the DAQ system
Flex: Provide connections to HV, LV, and data transfer to and from the module.

The location of the pixel modules is highlighted in green



The ITk

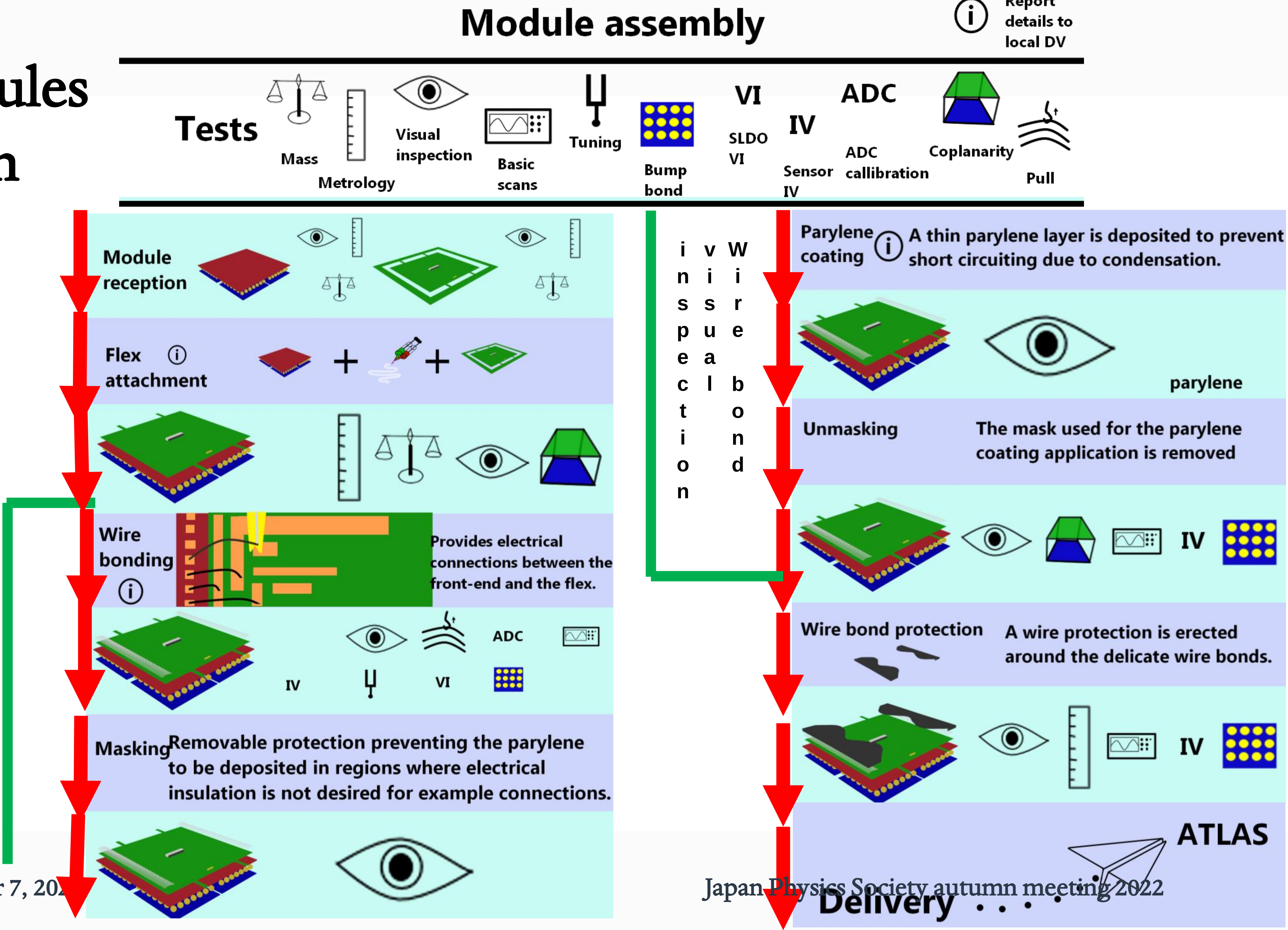




Pixel modules production

Japan will oversee the production and testing of the quad pixel modules which amount to about 2200 of the 10000 pixel modules.

Visual inspection



Motivation

During the production, the automatization of the wire bond visual inspection can significantly reduce the workload

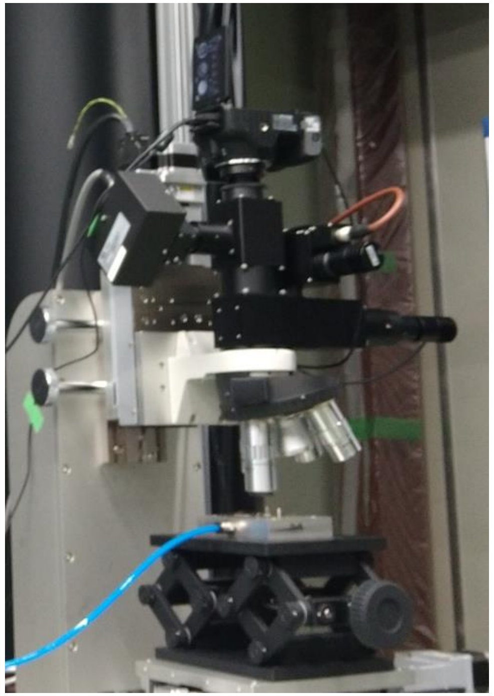
669 wires per module × 2,200 modules × 4 visual inspections=5,887,200 wire visual inspections

5,887,200 wire visual inspections × 1 inspection/second ≈ 1635.33hours≈204.4 working days ≈9.54 months

Objectives

- Automate the wire bond visual inspection.
- Simplify the image capturing process so that it can be done with minimal training.
- Detect and highlight broken, loosen, or touching wires.
- Develop a general wire bond visual inspection system capable of not only working on the pixel modules but also having the potential to be adapted for other future detectors.

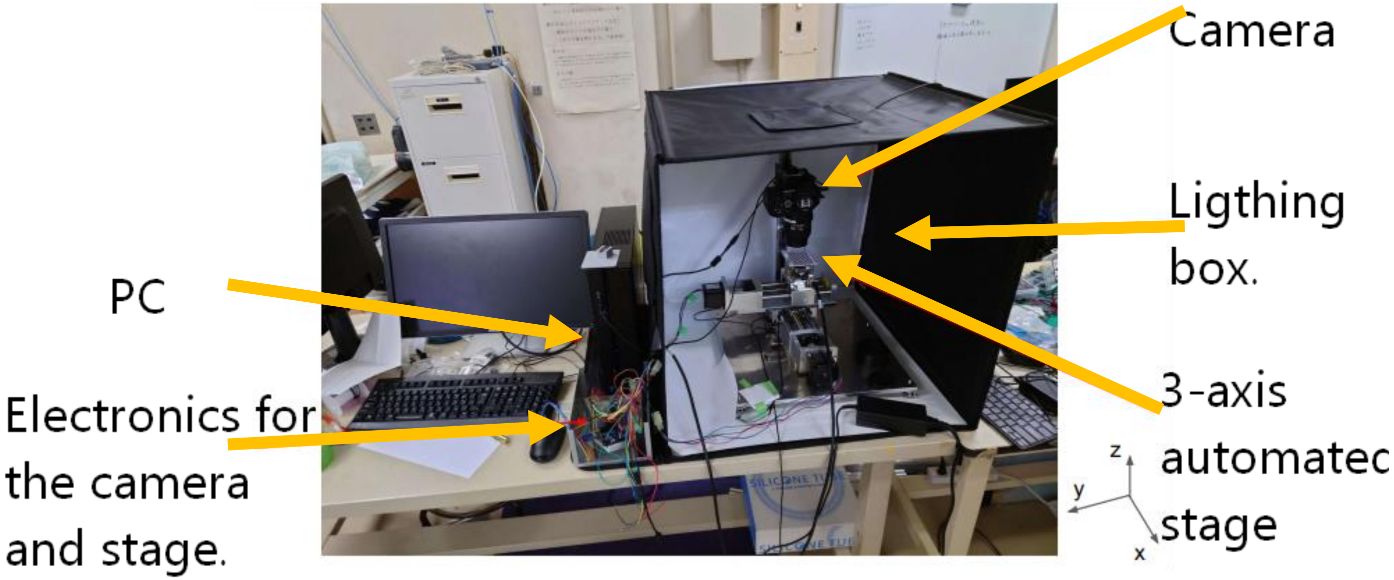
The capturing devices



The metrology machine

Is a fully automated microscope used for measuring the height and flatness of the module.

Advantages: Well-tested, very precise, and versatile.
Disadvantage: Very busy with metrology.
The modules need to be removed from the carrier.



The visual inspection machine

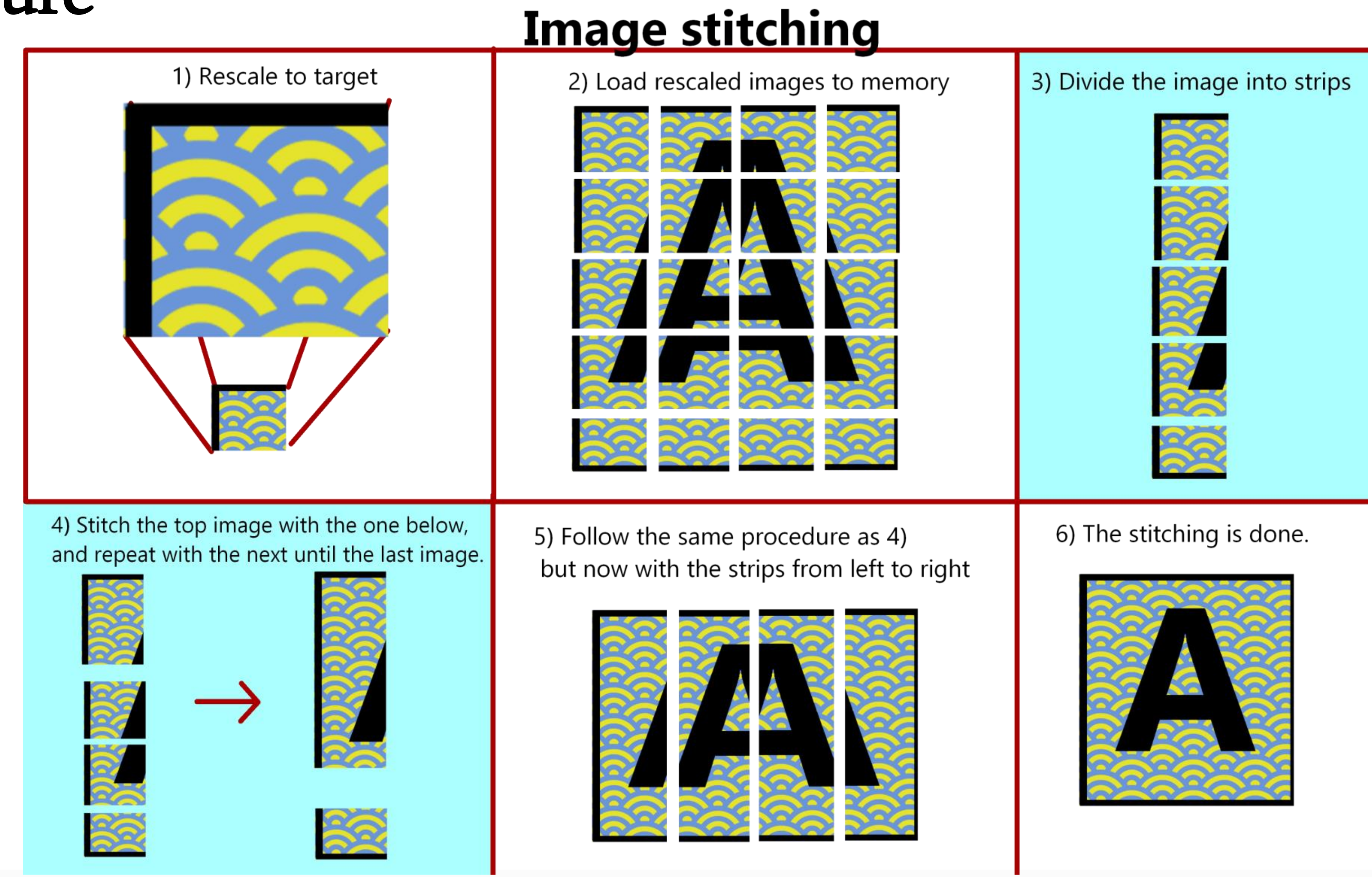
Developed by Jinnouchi laboratory, consists of a camera and an automated stage.

Advantages: Modules can be placed with the carrier on.
Can be modified to better suit the needs of the visual inspection.
Disadvantage: Not as precise as the metrology machine.

The image capture

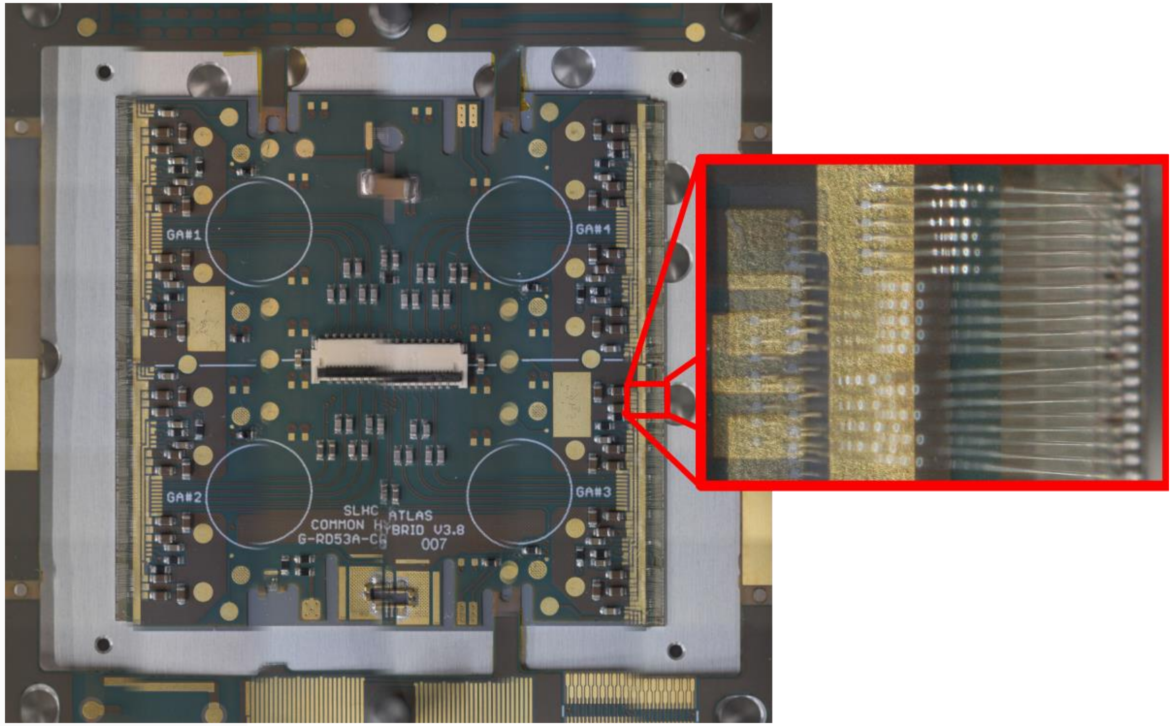
A visual inspection consists of 36 images of 4000×600 image pixels.

The images are stitched together into a single image of 25000×25000 image pixels.



Optimal stitching

- The stitching algorithm has already been implemented in the ATLAS team for visual inspection and some metrology.



Without the optimization, the images will not overlap perfectly.

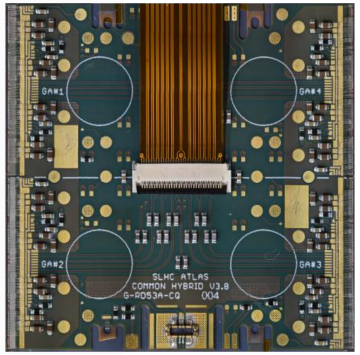
The perfect stitching can be obtained by optimizing the 6 parameters of the transformation (homography plus translation)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} H_{xx} & H_{xy} \\ H_{yx} & H_{yy} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

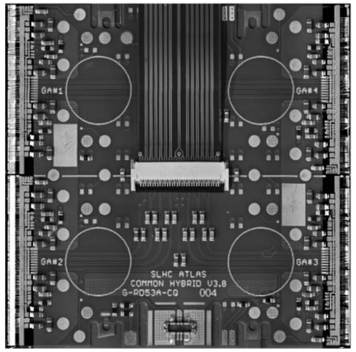
So that the cost function of the overlapping is minimized. There are many cost functions like GIST1, GIST2, and the image difference.

Wire bond visual inspection

For the automated wire bond visual inspection **Steger's curve tracking algorithm** will be used. The algorithm has already been presented at the 2022 spring JPS meeting talk A561 Room 17aA561.

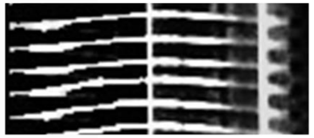


Standarized image



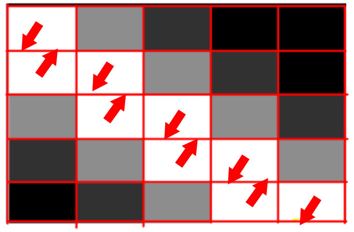
To grayscale

Steger's curve tracking algorithm

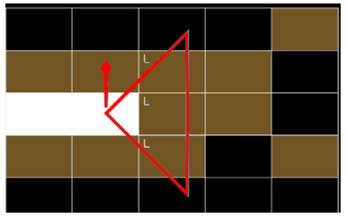


Gaussian blur

Obtain the eigen-values and eigen-vectors of the Hessian matrix



Linking algorithm



Analysis

Looking for broken wires

Is the curve continuous from A to B?

Looking for loose wires or incorrect connections

Is the end of the curve close to A?

Looking for loose wires or incorrect connections

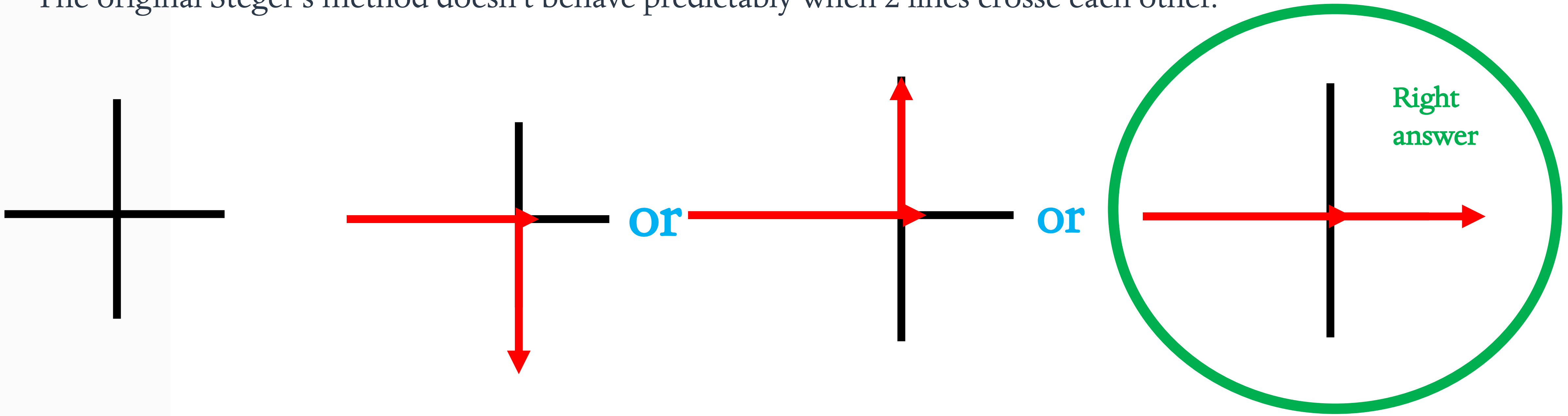
Is the end of the curve close to B?

Looking for shortcircuits

Is the curve too close to its neighbors?

Kalman stabilizer

The original Steger's method doesn't behave predictably when 2 lines crosse each other.



Example
Which line is which?

Steger's turn right about
33% of the times

Steger's turn left about
33% of the times

Steger's goes straight about
33% of the times

After adding a Kalman filter to the Steger's linking algorithm the problem is solved.

Wire bond quality analysis

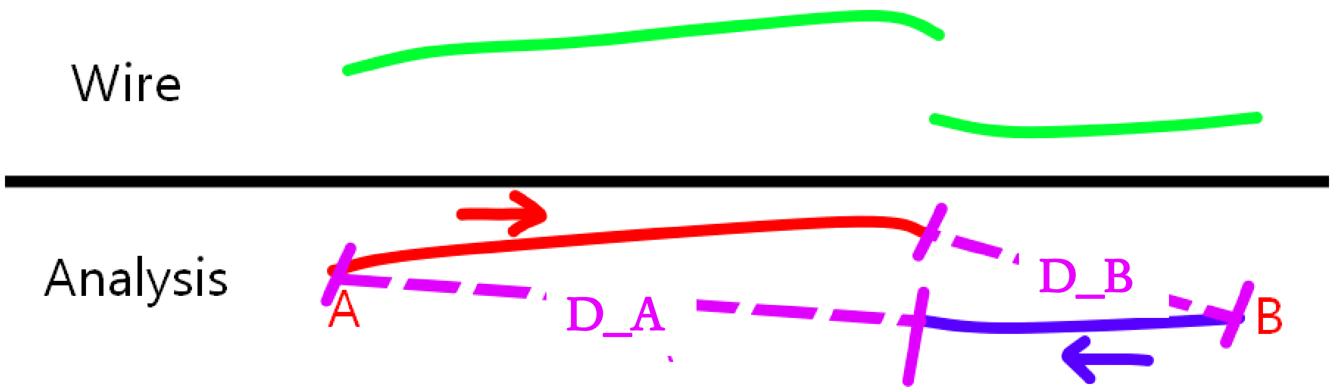
For the analysis the wires are tracked twice, once from the left to the right and another from the right to the left

Defective wire bonds are divided into 3 categories:

A) Broken wires
B) Touching wires
C) Loose wire

Wire bond quality analysis

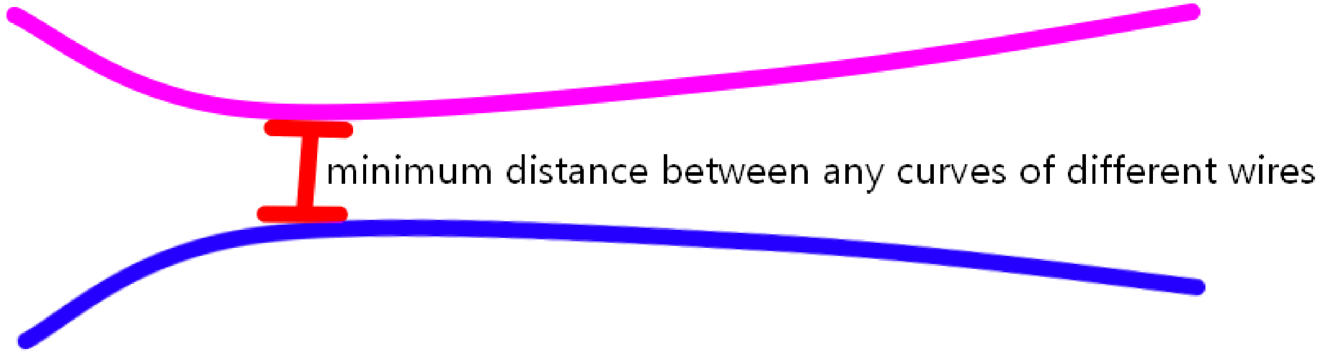
*The red line represents the tracking from left to right starting in A
The blue line represents the tracking from right to left starting in B



* D_A and D_B are the distance between the end of the tracked line and the target point A or B

A) Broken wires: A broken wire is detected if D_A and D_B are bigger than one wire width.

B) Touching wires:
If any two image pixels belonging to different wires get closer than 2.5 wire widths the wires are labeled as potentially in short circuit.



Wire bond quality analysis

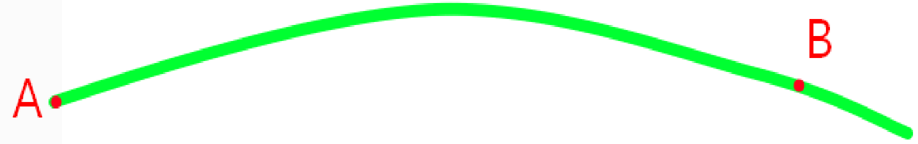
*The red line represents the tracking from left to right starting in A
The blue line represents the tracking from right to left starting in B

C) Loose wire = One end of the wire is not connected (the soldering failed) :

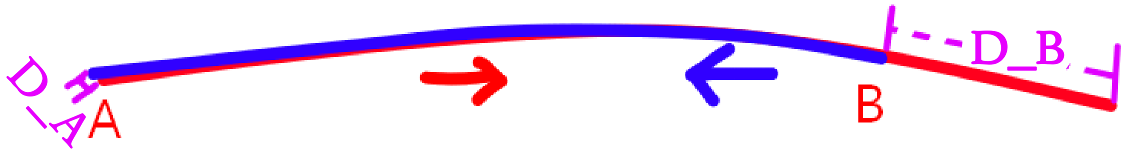
There are 2 possible cases

- Case 1: A loose wire is defined if D_A or D_B but not both are bigger than one wire width

Wire case 1

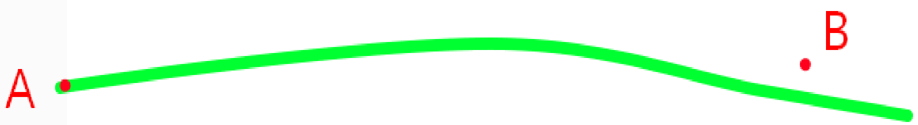


Analysis



- Case 2: A loose wire condition is declared if only one of the 2 trackings succeeded but the other failed immediately or after only a few pixels

Wire case 2



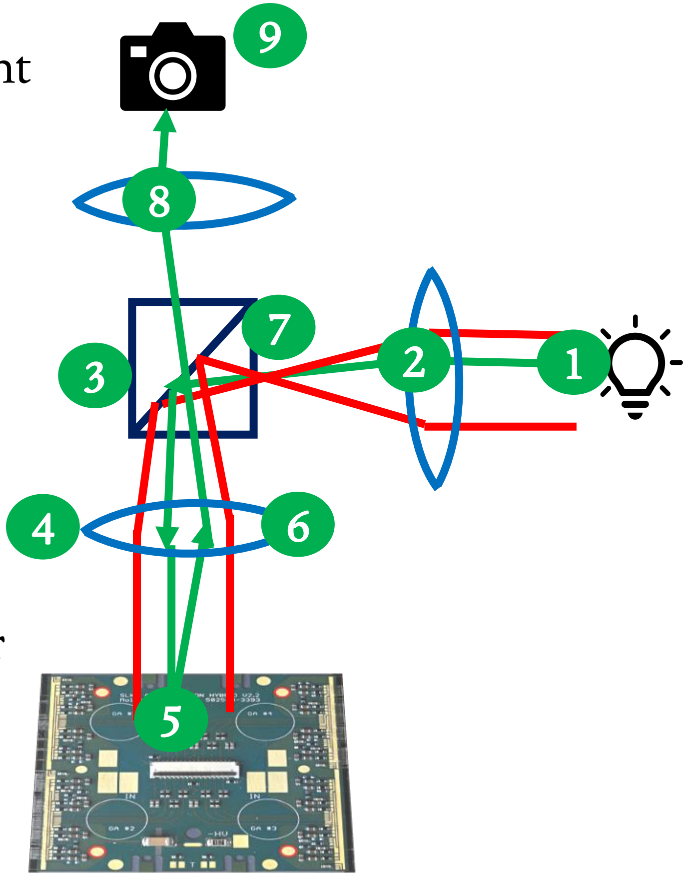
Analysis

Lighting conditions

Coaxial lighting

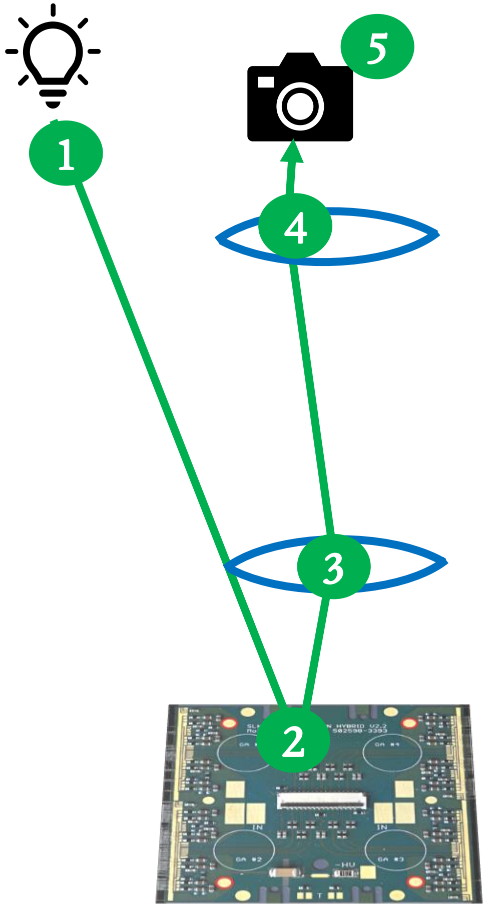
Only available on the metrology machine

1) Collimated light source
2) Lens
3) Beam splitter
4) Objective
5) Sample
6) Objective
7) Beam splitter
8) Ocular
9) Camera sensor



Ambient lighting

Does not specify the light source



1) External light sources
2) Sample
3) Objective
4) Ocular
5) Camera sensor

Overall defective wires detection rate (broken or loose)

Results

Simulated images (Test 1, Test 2) — Real world images (Test 3, Test 4)

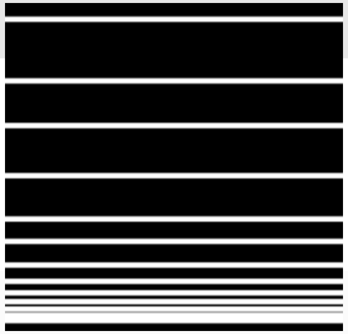
Percentage of good wires identified as with no defects

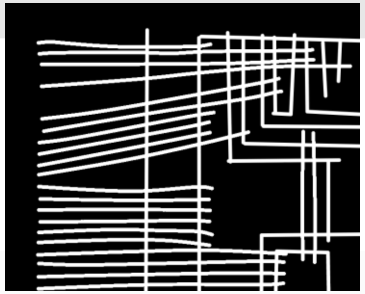
	Test 1 (easiest)	Test 2 (easy)	Test 3 (moderate)	Test 4 (difficult)
Sample size	12	50	50	50
Steger's wire tracking with Kalman filter	92%	98%	34%	10%

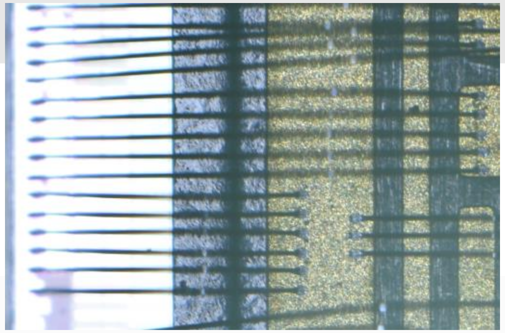
Percentage of defective wires identified as defective

*NA means does not apply

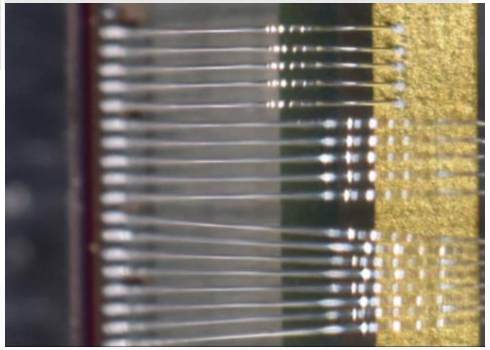
	Test 1 (easiest)	Test 2 (easy)	Test 3 (moderate)	Test 4 (difficult)
Sample size	0	7	0	7
Steger's wire tracking with Kalman filter	NA	100%	NA	100%







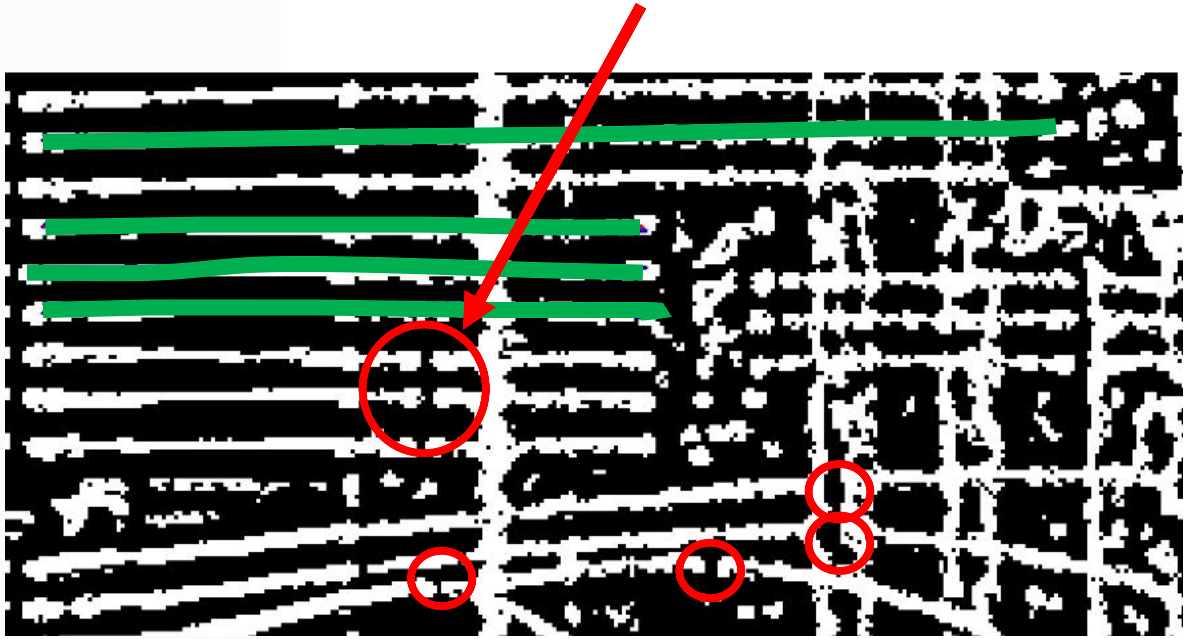
Coaxial lighting



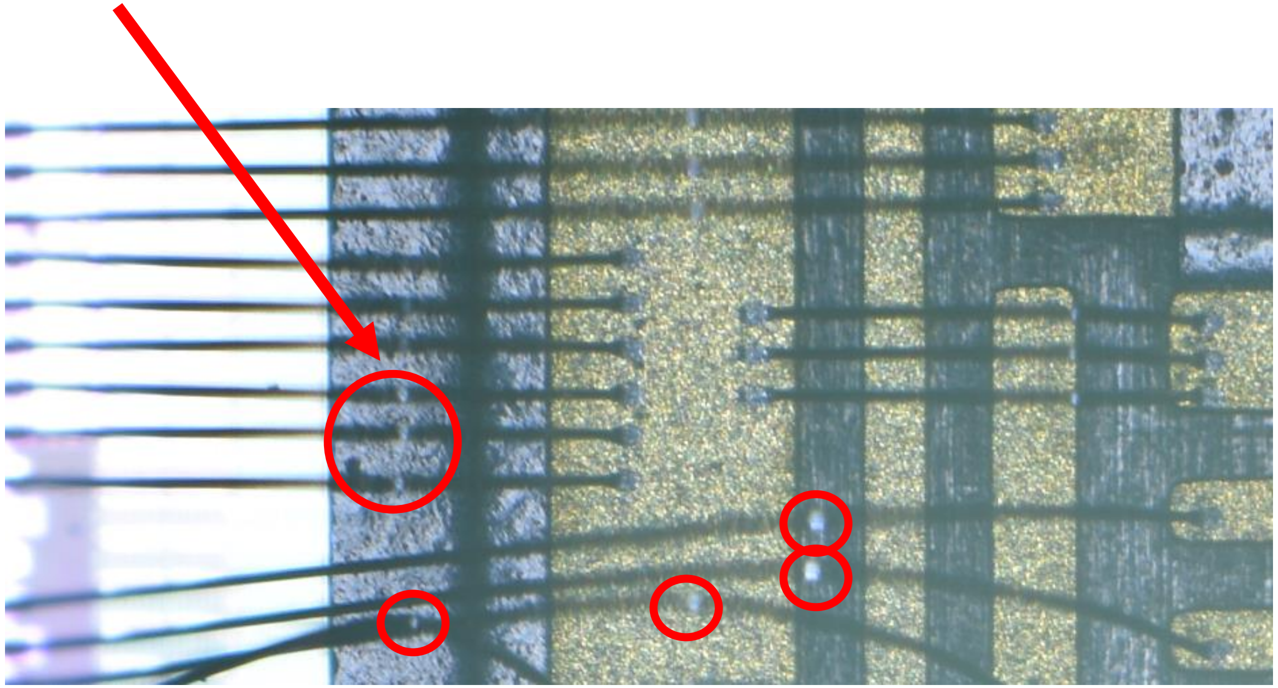
Ambient lighting

Discussion and future work

- On tests 3 and 4 the linking algorithm stops prematurely because the lines **appear broken** due to the **specular reflection** on the wires. This problem needs to be addressed.



The image pixels on the center of the line according to Steger's algorithm are represented in white. In blue and red the wire tracked



Location of the missing image pixels on the original image
(Test 3 Coaxial lighting).

Conclusion

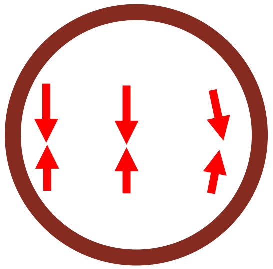
- The capture of the visual inspection has been simplified with the incorporation of the image stitching algorithm.
- A technique for automatically identifying defective wires has been developed.
- By implementing the wire tracking algorithm, the workload is expected to be reduced by more than 90% (based on tests 1 and 2).
- The difficulties found on tests 3 and 4 are due to problems with the specular reflection on the wires. Solutions are being studied for mitigating the problem in the future, once the problem has been solved the algorithms presented should have no problem tracking the wires. At the moment even with the specular reflections, the workload would be reduced by around 34% (with coaxial lighting) and 10% (under ambient lighting).
- All the damaged wire examples were detected correctly. (~100% detection rate)

Backup

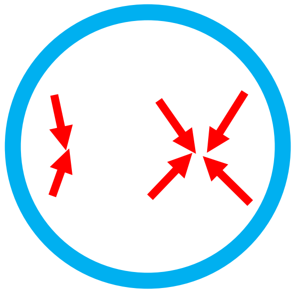
The Kalman filter application

Uncertainty eigenvector = standard deviation

Measurements
- Position x
- Position y
- Eigenvalue
- Eigenvector x component
- Eigenvector y component



Eigenvectors on an
Independent line



Eigenvectors near
the intersection of 2 lines

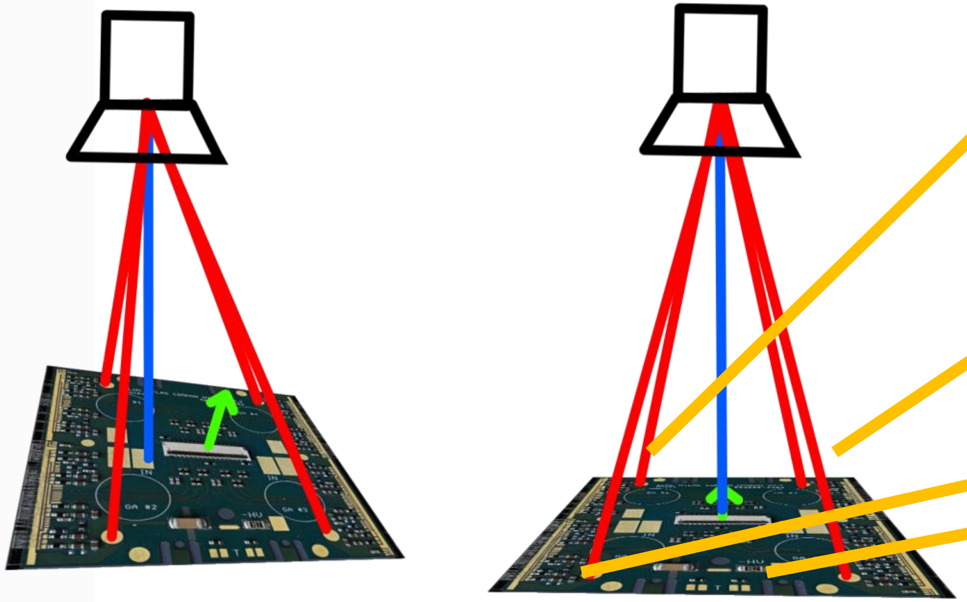
Measurements take priority

Predictions take priority

Image standardization

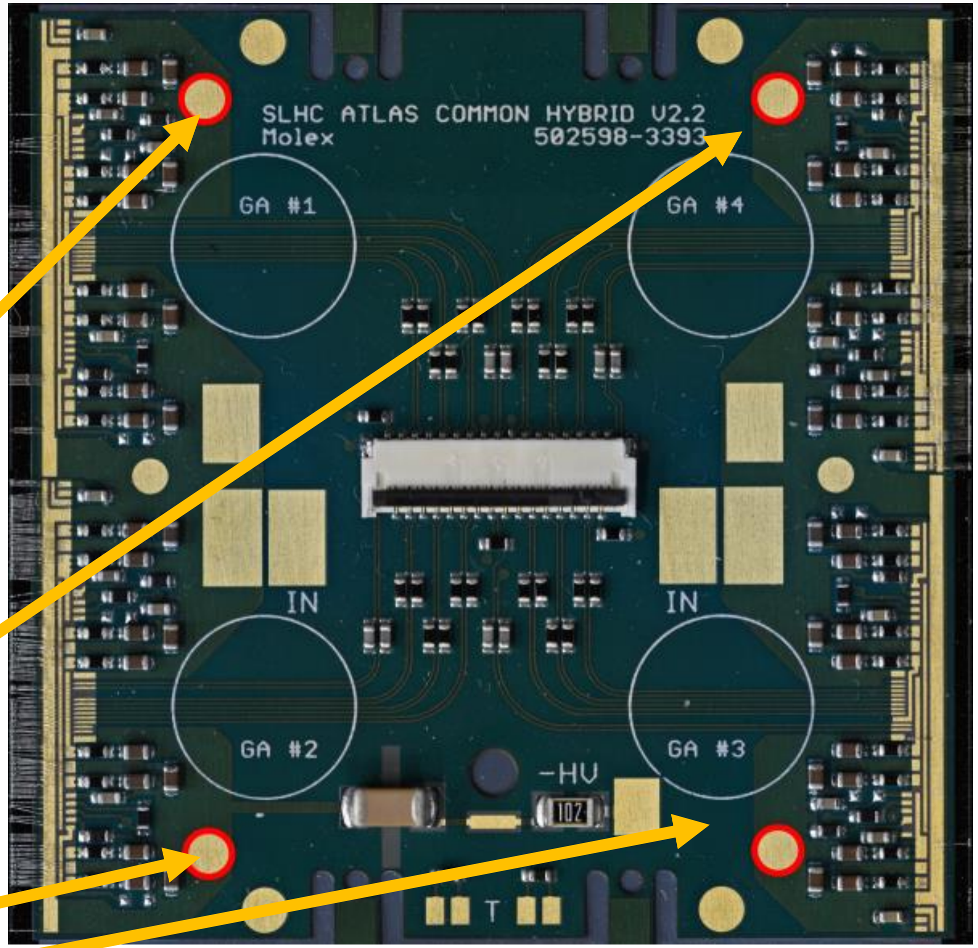
The image produced by the stitching is transformed by a (homography plus translation) to the desired size, perspective, and position.

The image after the transformation has a **known mapping** between image coordinates in pixels and coordinates in the flex



a) Without correction

b) After correction



For the mapping, the circles indicated in red are used as **reference points**.

The output image is 10000 × 10000 image pixels in size

Cost functions of the overlap

The image difference: Calculates the difference

$$d(f,g) = \sum\sum_{(x,y)\in A\cap B} |a(x,y) - b(x,y)|_p$$

With: $$d(f,g,W,R) = \sum \sum_{(x,y)\in R} W(x,y)|a(x,y) - b(x,y)|_p$$

GIST1:

$$\begin{aligned} \mathrm{GIST1}(A,B) &= d(\nabla A, \nabla C, 1-W, A\cap B^c) + d(\nabla A, \nabla C, 1-W, A\cap B) \\ &\quad + d(\nabla B, \nabla C, W, A\cap B) + d(\nabla B, \nabla C, W, A^c\cap B) \\ &= d(\nabla A, \nabla C, 1, A\cap B^c) + d(\nabla A, \nabla C, 1-W, A\cap B) + d(\nabla B, \nabla C, W, A\cap B) + d(\nabla B, \nabla C, 1, A^c\cap B) \end{aligned}$$

GIST2:

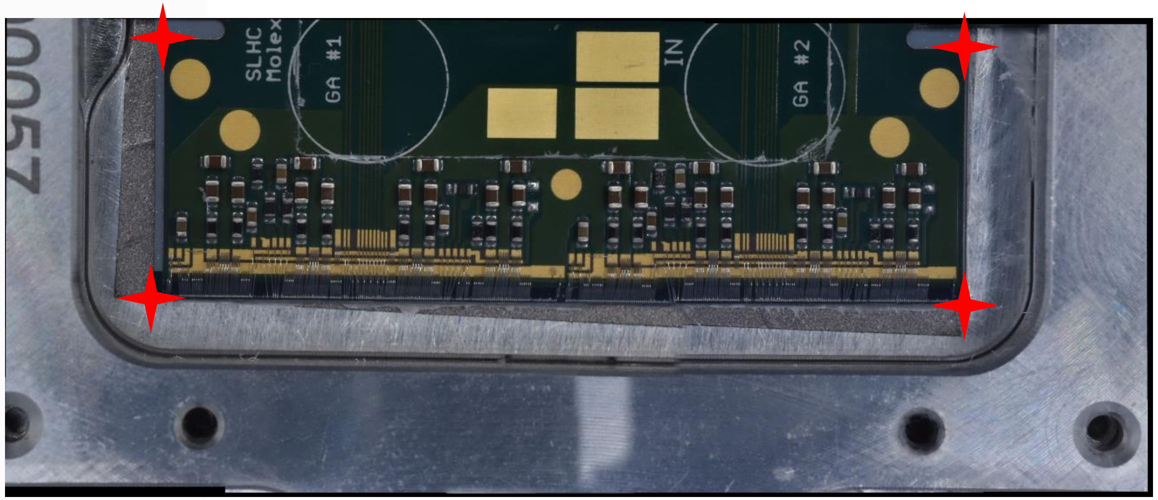
$$\begin{aligned} GIST2(A,B) &= \sum \sum_{(x,y)\in A\cup B} |\nabla S(A,B) - (S(\frac{\partial A}{\partial x}, \frac{\partial B}{\partial x}), S(\frac{\partial A}{\partial y}, \frac{\partial B}{\partial y}))|_p \\ &= \sum \sum_{(x,y)\in A\cup B} |(\frac{\partial S(A,B)}{\partial x} - S(\frac{\partial A}{\partial x}, \frac{\partial B}{\partial x}), \frac{\partial S(A,B)}{\partial y} - S(\frac{\partial A}{\partial y}, \frac{\partial B}{\partial y})|_p \end{aligned}$$

Perspective correction (Homography)

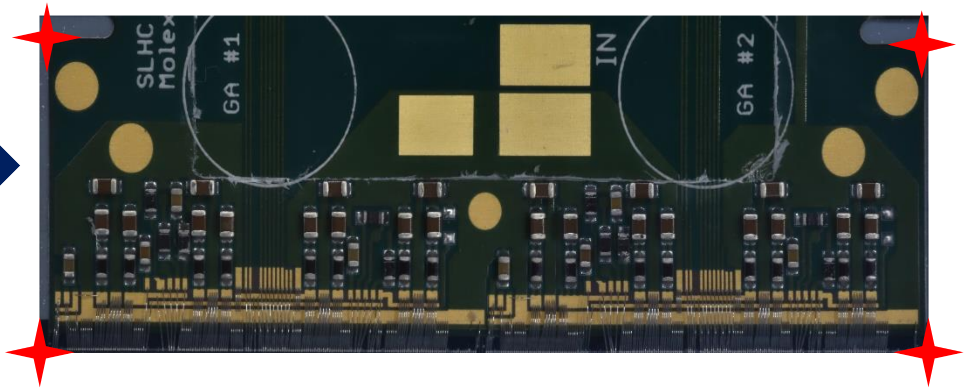
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

A homography is a transformation between 2 planes and has 8 degrees of freedom

Is usually determined by 4 markers (references) in an image and their corresponding transformation

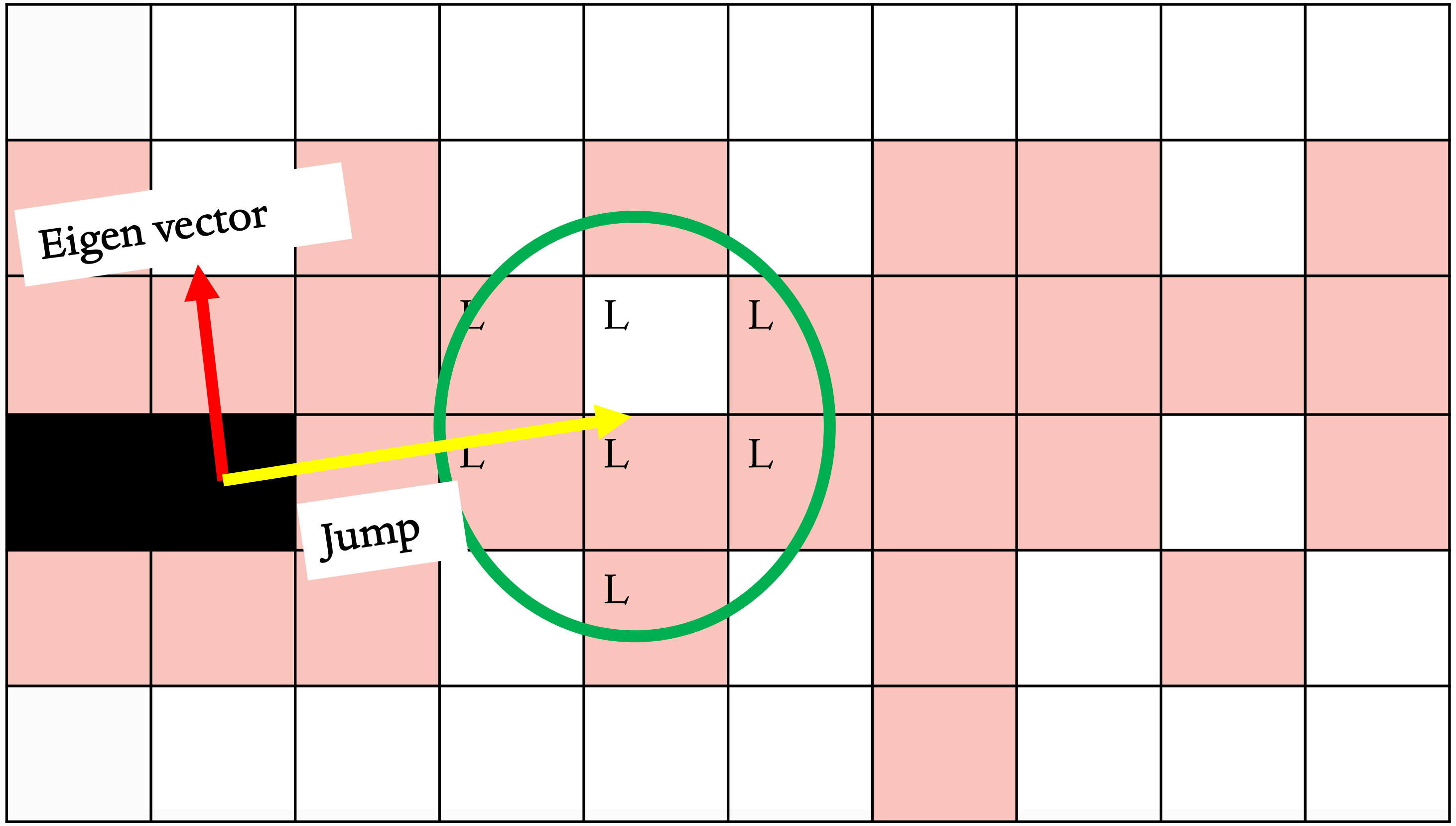


H



*This image was also trimmed to a standardized size

Jump Linking algorithm (Modified linking algorithm)



Pixel non in the center of the line (non Steger's Pixel)

Steger's pixel

Look up area. (Any kernel shape is possible).

L Pixels inside the lookup area

Look up around that point for the next pixel , the cost function is $C = (r - r_0) + W_\alpha(\alpha - \alpha_0) + W_{ev}(ev - ev_0)$
Where r are the coordinates of the pixels, α the angle of the eigen vector and ev the eigen value.

Kalman filter (stabilizer)

Instead of a single measurement of the position and eigenvector we could use previous measurements to improve our estimation, specially in noisy regions or in the intersections of two lines.

On this work the Kalman filter tracks the state vector S which a point in the curve at time t

$$S(t) = \begin{bmatrix} X(t) \\ P(t) \\ ev(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \\ u(t) \\ v(t) \\ ev(t) \end{bmatrix}$$

Position

Eigenvector (normalized)

eigenvalue

Kalman filter (stabilizer)

Let be s the state of the system at time t, the state at time $t + \Delta t$ should be:

$$s_{t+\Delta t} = s_t + \frac{\Delta s}{\Delta t}\Delta t$$

Kalman propose we can estimate $\frac{\Delta x}{\Delta t}$ as a linear function of the previous state (prediction Fx_t) plus a function of our measurement $u_{t+\Delta t}$ obtained at time $t + \Delta t$ plus a constant :

$$\frac{\Delta s}{\Delta t} = Fs_t + Gu_{t+\Delta t} + w$$