DIBRIS

DEPARTMENT OF INFORMATICS,
BIOENGINEERING, ROBOTICS AND SYSTEM ENGINEERING

COMPUTER VISION

# Second Assignment
## Computer Vision Laboratory Report


*Author :*

Bahri Riadh
Ruggeri Edoardo
Tozzola Michele

*Student ID :*

S8335614
S7782369
S8995647

*Professors :*

Solari Fabio
Nicoletta Noceti

*Tutor :*

Razeen Hussain


21 novembre 2025

# GENERAL INTRODUCTION

The main goal of this laboratory is to work with two important computer vision techniques : template matching and corner detection.

1. The first part focused on using Normalized Cross-Correlation (NCC) to track two cars : a red one and a dark one, across six consecutive images. By selecting a small image patch (a template) and comparing it with each frame, it is possible to locate the cars and study how the size of the template affects both the accuracy and the processing time.

2. The second part of the lab was about implementing the Harris Corner Detector, a classic method used to find interest points in an image. Image gradients were computed, Gaussian smoothing was applied, the Harris response map was constructed, and the strongest corner points were finally detected and visualized.

Overall, this lab provided hands-on experience with fundamental techniques used in many computer vision applications, including object tracking, feature extraction, and image analysis.

Question

# 1 NCC-based Segmentation

## 1.1 Introduction

In the first part of the lab, we used normalized cross-correlation (NCC) to track two vehicles across six grayscale frames. Templates of the red car and the dark car were extracted from the first image and matched against the remaining frames. We also evaluated how template size affects detection accuracy and computation time, and compared the results with the color-based segmentation method from Lab 4.

## 1.2 Results and Interpretation

### 1.2.1 Template Selection

To perform NCC-based segmentation, two templates were manually extracted from the first image (*ur_c_s_03a_01_L_0376.png*) :

- **Red car template** was taken from the region $(360\!:\!430,\ 690\!:\!770)$.
- **Dark car template** was taken from region $(370\!:\!410,\ 560\!:\!640)$.

These templates represent the appearance of the cars and serve as the reference for NCC.

### 1.2.2 NCC Score Maps and Detection

For each template and for each of the six frames, the normalized cross-correlation was computed using `normxcorr2`. The resulting NCC score map highlights the regions of the image that best match the template. The maximum of the score map indicates the most likely correspondence, and a bounding box with the same dimensions as the template is drawn at this location.

### 1.2.3 Red Car Detection

The red car was detected consistently across all frames as shown in the figure 1.1a.

$*$ **Observations :**

- NCC produced a clear and stable peak in every score map
- The estimated bounding box aligned accurately with the car

$*$ **Interpretation :**This is expected, since the red car remains static in position, orientation, and scale.

(a) Detected red car (b) Detected dark car

FIGURE 1.1 – Cars detection

### 1.2.4 Dark Car Detection

The dark car is more challenging because it moves, rotates, and slightly changes scale while turning left (see figure 1.1b).

* **Observations :**
  - In early frames, the match is accurate and stable
  - As the car rotates, the NCC peak becomes broader but still identifies the correct location
  - Minor drift appears in some frames, but overall detection remains acceptable
* **Interpretation :** This highlights the sensitivity of NCC to scale and rotation variations, which do not affect the red car detection.

### 1.2.5 Comparison with Lab 4 (Color-Based Segmentation)

* **Robustness and Limitations :**
  - **HSV :** Sensitive to environmental factors such as shadows and overlapping colors, which can affect accuracy.
  - **NCC :** Robust against intensity variations (lighting changes), but it struggles with geometric changes such as scale, rotation, or partial occlusion.
* **Computational Efficiency :**
  - **HSV :** Superior efficiency, as it mainly involves generating a threshold-based mask.
  - **NCC :** Computationally intensive. Execution time increases with template size and image quality (see chapter 1.2.6.1).
* **Experimental Results :**
  - **HSV Performance :** Successfully detected the black car in all six frames. However, reducing the template size caused detection to shift to a parked black car due to similar color characteristics.
  - **NCC Performance :** Delivered precise and accurate results. It performed well in this scenario because the target did not undergo significant rotation or size variations.

⟹ Overall, the choice between HSV segmentation and NCC depends on the nature of the object and the variability of the environment : HSV is ideal for color-based scenarios, while NCC is better suited for shape or texture-based detection.

### 1.2.6 Different sizes of template

In this part of the study, three template sizes were compared. All templates were centered on the black car and were obtained by scaling the original template by factors of $\times 2$, $\times 4$, and $\times 0.2$. The comparison focuses on **computation time** and **detection accuracy**.

#### 1.2.6.1 Computation time

In NCC, the cross-correlation operation is performed for every possible position of the template on the target image ; therefore, the computational load is strongly affected by the size of the template. Larger templates require more pixel comparisons, which increases the processing time due to the added complexity of the calculations.

| Size | Mean time (s) |
|---|---|
| $\times 0.2$ | 1.245 |
| $\times 2$ | 1.924 |
| $\times 4$ | 2.885 |

**TABLE 1.1 – Computation time for different template sizes**

**Table 1.1** shows the average computation times for the three template sizes, each representing the time required to compute the NCC over all six frames. The smallest template yields the shortest time, while the largest is the slowest, as expected from the increasing number of pixels processed.

By comparing the values, it is possible to notice that the increase in computation time is not linear with respect to the template size. Although the size difference between the first two templates is much larger than between the second and the third, the increase in computation time is greater between the second and the third sizes. This confirms that the relationship between template size and processing time is proportional but non-linear.

#### 1.2.6.2 Detection accuracy

Changing the size of the template strongly affects the accuracy of the detection.


**FIGURE 1.2 – Template matching result with a template size of x0.2**

* **Small Template** ($\times 0.2$ **Scale) (figure 1.2) :**
  - **Performance :** Correctly identifies the target in the first two frames and the fourth frame, but aligns with unrelated objects in the remaining frames.
  - **Cause of Failure :** The down-scaling removes distinctive features, resulting in limited visual information and decreased discriminative capability.
  - **Result :** The NCC algorithm produces misleading correlation peaks in areas that share only weak similarities with the original object.
* **Intermediate Template** ($\times 2$ **Scale) :**
  - **Performance :** Successfully tracks the black car in all frames by capturing the entire object along with a small margin.
  - **Computational Note :** The primary difference compared to the original template is the computational time (the mean time for the original template is 1.327 s).

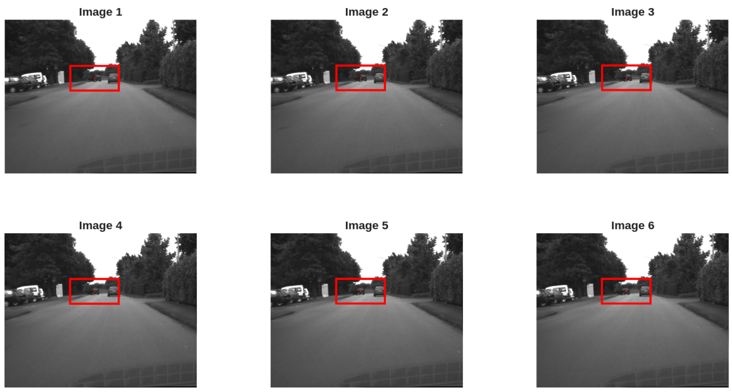



**FIGURE 1.3 – Template matching result with a template size of x4**

* **Large Template** ($\times 4$ **Scale)(figure 1.3) :**
  - **Performance :** Detects the car initially but fails to track the movement ; it remains stationary when the car turns left.
  - **Cause of Failure :** The template includes a significant amount of background data.
  - **Result :** The background produces a stronger response than the target during cross-correlation, causing the match to lock onto the background rather than the car.
* **Conclusion :**
  - **Optimal Balance :** The intermediate size ($\times 2$) offers the best trade-off between computation time and detection accuracy.
  - **Key Takeaway :** A robust template should encompass the entire object while excluding unnecessary background regions.

## 1.3 Conclusion

This chapter showed how NCC can be used to track objects across multiple frames. The red car was consistently detected, while the dark car highlighted the method's sensitivity to rotation and scale changes. The comparison with HSV segmentation confirmed that each technique is effective under different conditions. Finally, the template size experiment demonstrated the trade-off between accuracy and computation time.

Question

# 2 Harris Corner Detection

## 2.1 Introduction

The goal of this part of the lab was to implement the Harris Corner Detector and apply it to the image *i235.png*. The detector identifies points where the image intensity changes strongly in multiple directions, making them reliable corner features.

The main steps of the implementation were as follows :

- compute the horizontal and vertical gradients using **Sobel filters**,
- form the derivative products and smooth them with a **Gaussian filter**,
- compute the **Harris response map** $R$,
- threshold the response using $0.3 \cdot \max(R)$ to isolate strong corner regions,
- extract corner centroids using **regionprops()**,
- and finally overlay the detected corners on the original image.

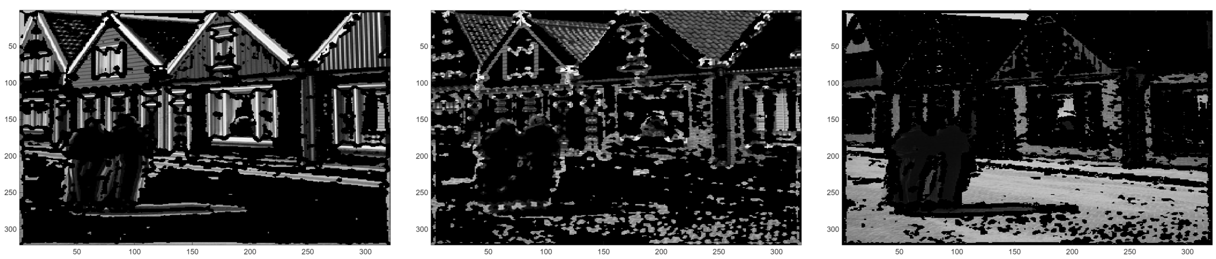
## 2.2 Results and Interpretation

### 2.2.1 Partial Derivatives (Ix and Iy)

To begin the Harris pipeline, we computed the partial derivatives $I_x$ and $I_y$ using Sobel filters. These derivatives highlight intensity changes in the image and provide the basic information required to identify corners.

### 2.2.2 Edge, Corner, and Flat Regions

Using the Harris response $R$, each pixel can be classified into edge regions (large negative $R$), corner regions (large positive $R$), or flat regions (values close to zero). These maps offer an intuitive interpretation of how the detector distinguishes different structures in the image.



**(a) Edge regions** **(b) Corner regions** **(c) Flat regions**

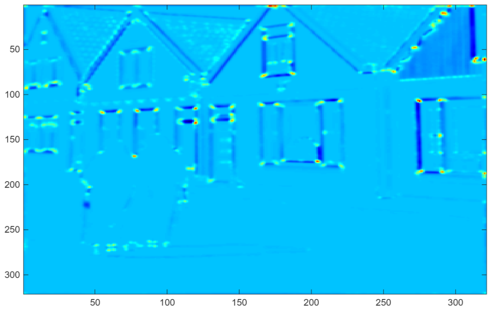
**FIGURE 2.1 – Classification of pixels into edge, corner, and flat regions**

- $*$ **Interpretation :**

Edges appear as long continuous structures (figure 2.1a), corner regions form compact bright spots (figure 2.1b), and flat regions dominate uniform areas (figure 2.1c).

### 2.2.3 Harris Response and Corner Regions

After smoothing the derivative products, we computed the Harris response $R$ for each pixel (figure 2.2a). Brighter values indicate stronger corner responses. To extract only the strongest candidates, we applied a threshold of $0.3 \cdot \max(R)$ (figure 2.2b).




**(a) Harris response map ($R$).**

**(b) Corner regions over the threshold**

**FIGURE 2.2 – Harris response and corner regions over threshold side-by-side.**

* **Interpretation :**

The $R$ map highlights pixels with strong cornerness. After thresholding, only the strongest responses remain, forming compact corner blobs from which centroids are extracted.

### 2.2.4 Final Detected Corners

The centroid of each corner region was computed and overlaid on the original image to obtain the final set of detected corners.

**FIGURE 2.3 – Detected corners on *i235.png*.**

* **Interpretation :**

Most detected corners lie at meaningful geometric structures such as edges, intersections, and textured regions. Some false positives are expected but can be reduced by tuning the threshold or Gaussian filter.

## 2.3 Conclusion

The Harris Corner Detector was successfully implemented and applied to the image *i235.png*. The results showed that the detector identifies well-localized interest points, especially at edge intersections and textured regions. Thresholding plays a crucial role in selecting only the strongest corner responses.

# GENERAL CONCLUSION

In this lab, we implemented and analyzed two foundational techniques in computer vision : Harris corner detection and NCC-based template matching. Through the Harris detector, we observed how corner features arise from strong intensity variations and how smoothing, gradient computation, and thresholding influence the final set of detected keypoints. This provided a practical understanding of how local image structure can be characterized and used in feature-based applications.

For NCC-based segmentation, we tracked two vehicles across multiple frames and examined how template size, object motion, and appearance influence detection quality. We also compared NCC with the color-based segmentation method from Lab 4, highlighting how the reliability of each approach depends on factors such as object color, lighting conditions, scale consistency, and background complexity.

Overall, this lab demonstrated the practical importance of choosing detection techniques that match the characteristics of the scene. It also emphasized how gradient-based features and template matching operate in real images, offering insight into their strengths, limitations, and appropriate use cases in computer vision tasks.