

Smart NV AI- Assisted Puzzle Solver

Reflection

Ibrahim Afkir
Student Bachelor Applied Computer Science

Table of contents

1. INTRODUCTION	3
2. SUBSTANTIVE REFLECTION	4
3. PERSONAL REFLECTION	6

1. Introduction

This document reflects on the internship carried out at Smart Toys and Games (Smart NV), a Belgian toy and game manufacturer, during the academic year 2025–2026. Over the course of the internship, the goal was to explore how computer vision could be used to enhance the physical puzzle experience for children. The main focus was on building an AI-assisted puzzle solver that detects physical puzzle pieces from a camera photo and provides real-time guidance to help the player continue solving.

This reflection is divided into two parts. The first part offers a substantive reflection on the work that was delivered, what it means for Smart NV, and what recommendations can be made for the future. The second part is a personal reflection on what this experience has meant in terms of professional growth, the challenges that came along the way, and the lessons that will carry forward beyond the internship.

2. Substantive reflection

The goal of this internship was to build a computer vision system that could detect physical puzzle pieces from a camera photo and provide real-time solving assistance. Looking back, not only was this goal achieved, but the project went beyond what was originally planned. The initial scope covered a single puzzle, IQ Waves, but by the end of the internship a second puzzle, IQ Stars, was also fully implemented, and a contribution was made to an augmented reality game the company was developing.

Both the IQ Waves and IQ Stars applications are fully functional. A user can open either app on their phone, scan the physical puzzle board, and receive progressive hints with voice guidance to help them continue solving. The app detects all pieces and the board using a YOLOv26 segmentation model, corrects the camera perspective, maps each piece to the digital grid, and runs a solver that checks solvability and generates hints. Everything runs entirely in the browser with no server, no installation, and no internet connection needed after the initial load. IQ Stars uses a completely different board with a hexagonal grid and star-shaped pieces, yet the same architecture powered both applications, confirming that the approach is reusable across different Smart NV puzzles.

Comparing the final result to the original project plan, many things changed along the way. The plan included a Python backend with FastAPI, but during the project it became clear that running everything in the browser would be faster, cheaper, and simpler to deploy. The board detection was originally planned using OpenCV, but after testing it proved unreliable under different lighting conditions, so the approach was changed to let the YOLO model detect the board alongside the pieces. The project plan also described bounding box detection, but segmentation was chosen instead because the puzzle pieces touch each other on the board and bounding boxes would overlap too much. These changes were not failures of planning, they were natural outcomes of going deeper into the project and finding better approaches through hands-on experience.

The idea behind this project was entirely new to Smart NV. The company had never explored computer vision or AI-assisted gameplay before, so this was something that was being explored together. Both applications are ready to be used, and the concept has proven that it works reliably in practice. However, since the idea is still new and Smart NV has over 200 puzzles in their portfolio, the company will need to study the case further and test it more before rolling it out widely. The natural next step would be to integrate the applications into Smart NV's main platform and add information about this feature to the puzzle packaging, so that customers know the digital assistance is available.

One recommendation for the future is to find a way to automate the annotation of real images. Currently, labeling real photos in Roboflow is done entirely by hand, which is the most time-consuming part of building a dataset for a new puzzle. If the trained model could be used to pre-annotate new images, with a human only correcting the

mistakes, this would significantly speed up the process of supporting additional puzzles. The overall lesson from this project is that the quality of the training data drives everything. When the data is good, the model trains well, and the rest of the pipeline follows naturally.

3. Personal reflection

This was not my first experience with computer vision or AI, but it was the first time I worked on a project of this scale in a professional environment with real deadlines, real stakeholders, and real expectations. That made it very different from academic projects where the consequences of failing are limited to a grade. Here, the company was counting on results, and that added a level of pressure that pushed me to grow in ways I did not expect.

The most difficult period of the internship came early on. Shortly after starting, the company assigned an urgent project to the AI team: an augmented reality game that involved both computer vision and AR built in Unity. For almost two weeks, I put my main puzzle project aside and tried to make progress on the AR side. But Unity and AR development were not something I had experience with, and despite the effort, there were no results to show. At the same time, my main project was still in the research phase with no clear direction yet. I felt stressed because I had nothing concrete to present to the company, and my supervisor's visit was approaching. Eventually, as a team, we decided to be honest with our mentor. We explained that AR was outside our expertise, that we had given it our best effort but could not deliver what was needed, and that our main project was falling behind. Our mentor understood completely. He asked us to focus on what we were good at, training a YOLO model for the game, and the company's own developers took over the AR integration. This was an important lesson in knowing when to speak up rather than struggling in silence, and in recognizing the difference between a challenge worth pushing through and one that is genuinely outside your scope.

Computer vision turned out to be one of the most challenging and rewarding fields I have worked in. Unlike web development or software engineering, where AI tools can help you build an application quickly, computer vision requires a much deeper understanding of what is happening behind the scenes. You need to understand the data, know how to generate it, know how to evaluate it, and be able to debug problems that no AI assistant can see for you. Throughout this project, one of the biggest skills I developed was debugging. I could use AI to help me write code, but AI could not look at a warped image and tell me that the grid was misaligned, or that a mask was bleeding into a neighbouring piece. That part was entirely on me. I learned to guide the AI tools by understanding the problem first and then directing them toward the right approach, rather than relying on them to figure it out.

This experience confirmed what I want to do going forward. As an AI developer, I can do what a software developer does, but a software developer cannot easily do what an AI developer does. That distinction feels meaningful, and it is something I want to continue building on. Computer vision is a field I want to stay in, and I am also interested in going deeper into natural language processing and AI agents, areas I have already explored in personal projects. The common thread is using AI to solve real-world problems, and that is exactly what this internship allowed me to do.

What I am most proud of is that I did not just finish one project, I finished two. Both IQ Waves and IQ Stars are fully working applications that solve a real problem for Smart NV. Seeing a child scan their puzzle, receive a spoken hint, place the piece, and watch the app celebrate with confetti is the kind of result that makes all the debugging, the failed experiments, and the late training runs worth it. This internship showed me what it takes to bring an AI project from an idea to a working product, and that experience is something I will carry with me into my career.