Intergalactic Riksbanken Chip Authenticator - README

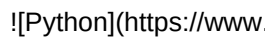
Group 8

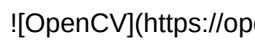
Suneela, Sara, and Abhishek

December 14, 2025

Intergalactic Riksbanken Chip Authenticator

> A computer vision system for authenticating and valuing intergalactic credit chips (Gold, Silver, Bronze) with simulator, camera, and interactive testing modes.





■ Quick Start

Launch All Modes (Recommended)

```
# Install dependencies
pip install -r requirements.txt

# Run the launcher
python launcher.py

# Select mode:
# 1 - Simulator Mode (conveyor belt)
# 2 - Camera Mode (real detection)
# 3 - Interactive Game (manual testing)
```

Individual Modes

Option 1: Simulator Mode (No Camera Required)

```
python main.py
```

Option 2: Camera Mode (Real-time Detection)

```
python camera_main.py
```

Option 3: Interactive Game (Manual Chip Testing)

```
python game.py
```

Features

Simulator Mode

• **Green Conveyor Belt**: 50% screen width, centered with moving texture

• **Three Chip Types**:

• **GOLD** (Yellow): Value = 3 digits × 10 (e.g., 752 → 7520 CR)

• **SILVER** (Blue): Value = 3 digits (e.g., 756 → 756 CR)

• **BRONZE** (Orange): Value = 2 digits × × (e.g., 2×4 → 8 CR)

• **Real & Fake Detection**: 80% real chips, 20% fake chips

• **Straight Line Movement**: Chips move perpendicular to belt motion

• **Real-time Statistics**: Track total value, real/fake counts

Camera Mode

Simulator Mode (`main.py`)

Key	Action
S	Spawn single chip
B	Burst spawn (5 chips)
C	Clear all chips
P	Pause/Resume
R	Reset statistics
Q	Quit

Camera Mode (`camera_main.py`)

Key	Action

Controls

Key	Action
S	Spawn single chip
B	Burst spawn (5 chips)
C	Clear all chips
P	Pause/Resume

■■■ camera_main.py # Camera detection system

R	Reset statistics
Q	Quit

Project Structure

```
chip_system/
■■■ launcher.py          # Main launcher (all modes)
■■■ main.py              # Conveyor simulator
■■■ camera_main.py       # Camera detection
■■■ game.py              # Interactive game
■■■ requirements.txt     # Dependencies
■■■ .gitignore          # Git ignore rules
■■■ README.md           # This file
■■■ QUICKSTART.md       # Quick reference
■■■ CHANGELOG.md        # Version history
■■■ LICENSE             # MIT License
■■■ CONTRIBUTING.md     # Guidelines
■■■ assets/             # Chip images
■   ■■■ gold.png
■   ■■■ silver.png
■   ■■■ bronze.png
■■■ docs/              # Documentation
    ■■■ CAMERA_USAGE.md
    ■■■ MIGRATION.md
```

System Requirements

- Python 3.8+
- OpenCV 4.8+
- NumPy 1.24+
- Webcam (optional, for camera mode)

Value Calculation Rules

Based on STB600 Final Project 2025:

Chip Type	Pattern	Calculation	Example
GOLD	3 digits	(d■×100 + d■×10 + d■) × 10	752 → 7520 CR
SILVER	3 digits	d■×100 + d■×10 + d■	756 → 756 CR
BRONZE	2 digits	d■ × d■	2×4 → 8 CR

Fake chips have 0 value and are marked in red.

Usage Example

```
from main import ConveyorSimulator

# Create simulator
sim = ConveyorSimulator(width=1280, height=720, conveyor_speed=3)

# Run
sim.run()
```

Credits

Created by Group 8:

- **Suneela**
- **Sara**
- **Abhishek**

STB600 Final Project 2025

Computer Vision & Image Processing