Using Generative AI To Increase Software Development Productivity

BECKHOFF

Who Am I?

BECKHOFF

Ryan O’Driscoll

- Mechanical Engineering Graduate of SJSU 24’
- Regional Support Engineer at Beckhoff Automation
- Previous Beckhoff Automation Intern of Summer 23’

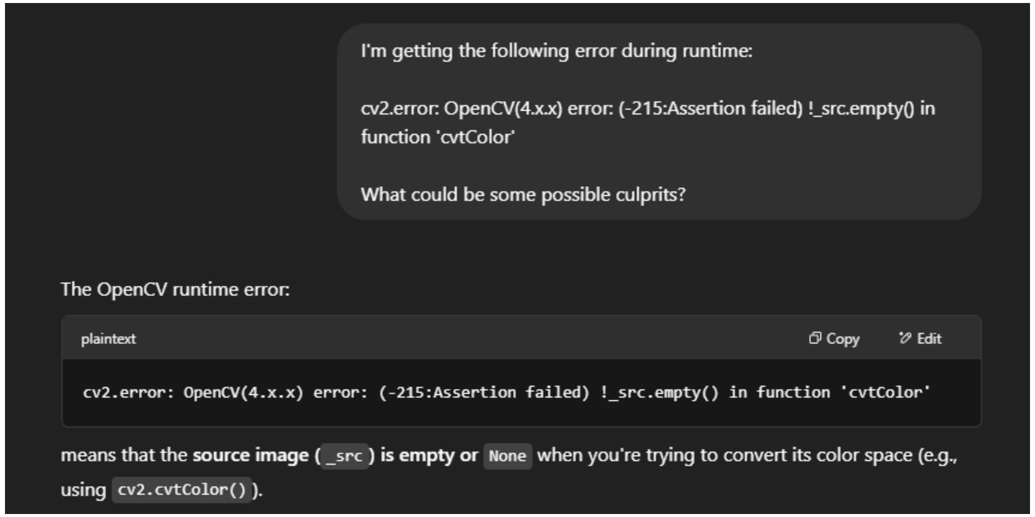
Internship Project:

- Built an XYZ Gantry that can play people in chess using chess engines and computer vision
- Used Beckhoff Motion control and Beckhoff’s ADS protocol to facilitate communication between computers
- Utilized LLM’s to complete the project in 4 weeks

How does AI come into play?

BECKHOFF

- Project took 4 weeks to complete from Concept ✉ First Prototype
  - Learning hurdles need to be overcome
  - Sometimes you need testing scripts
  - Compile/Runtime errors galore
  - All of these barriers in projects take time



I'm getting the following error during runtime:

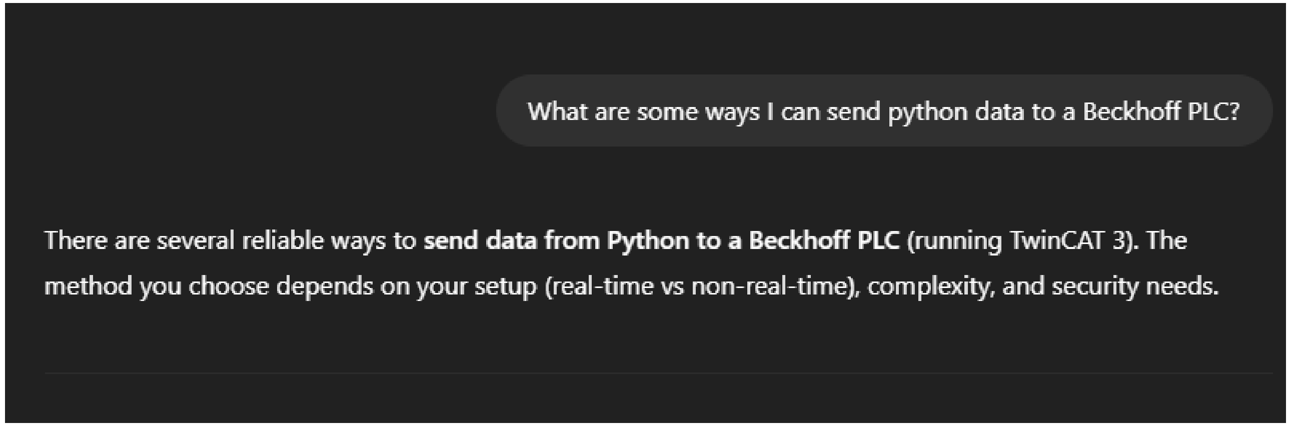
cv2.error: OpenCV(4.x.x) error: (-215:Assertion failed) !_src.empty() in function 'cvtColor'

What could be some possible culprits?

The OpenCV runtime error:

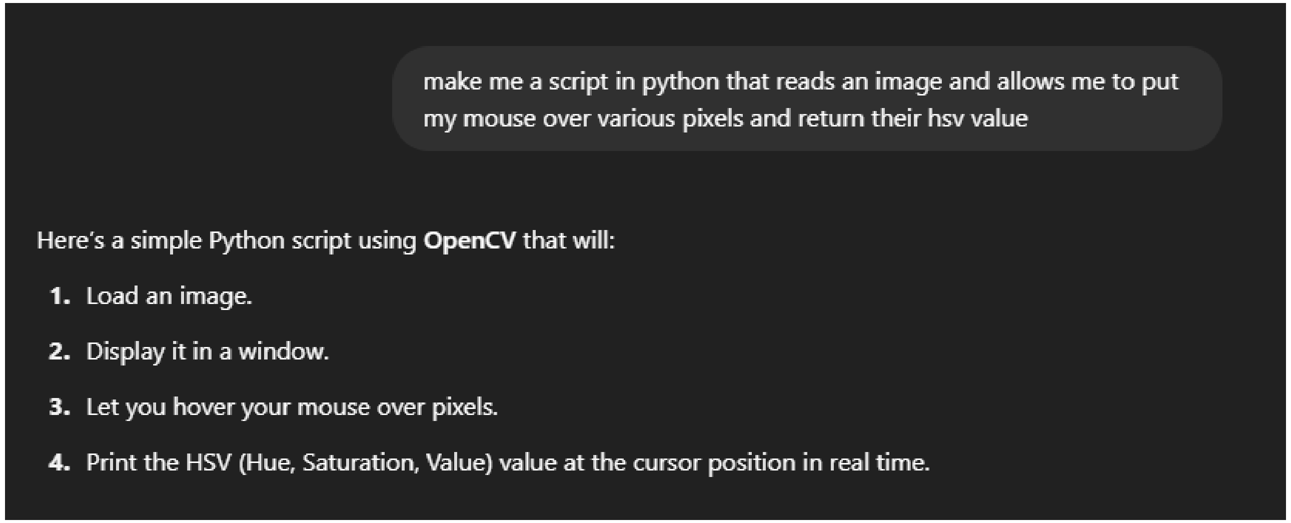
```
cv2.error: OpenCV(4.x.x) error: (-215:Assertion failed) !_src.empty() in function 'cvtColor'
```

means that the **source image** (`_src`) **is empty or** `None` when you're trying to convert its color space (e.g., using `cv2.cvtColor()`).



What are some ways I can send python data to a Beckhoff PLC?

There are several reliable ways to **send data from Python to a Beckhoff PLC** (running TwinCAT 3). The method you choose depends on your setup (real-time vs non-real-time), complexity, and security needs.



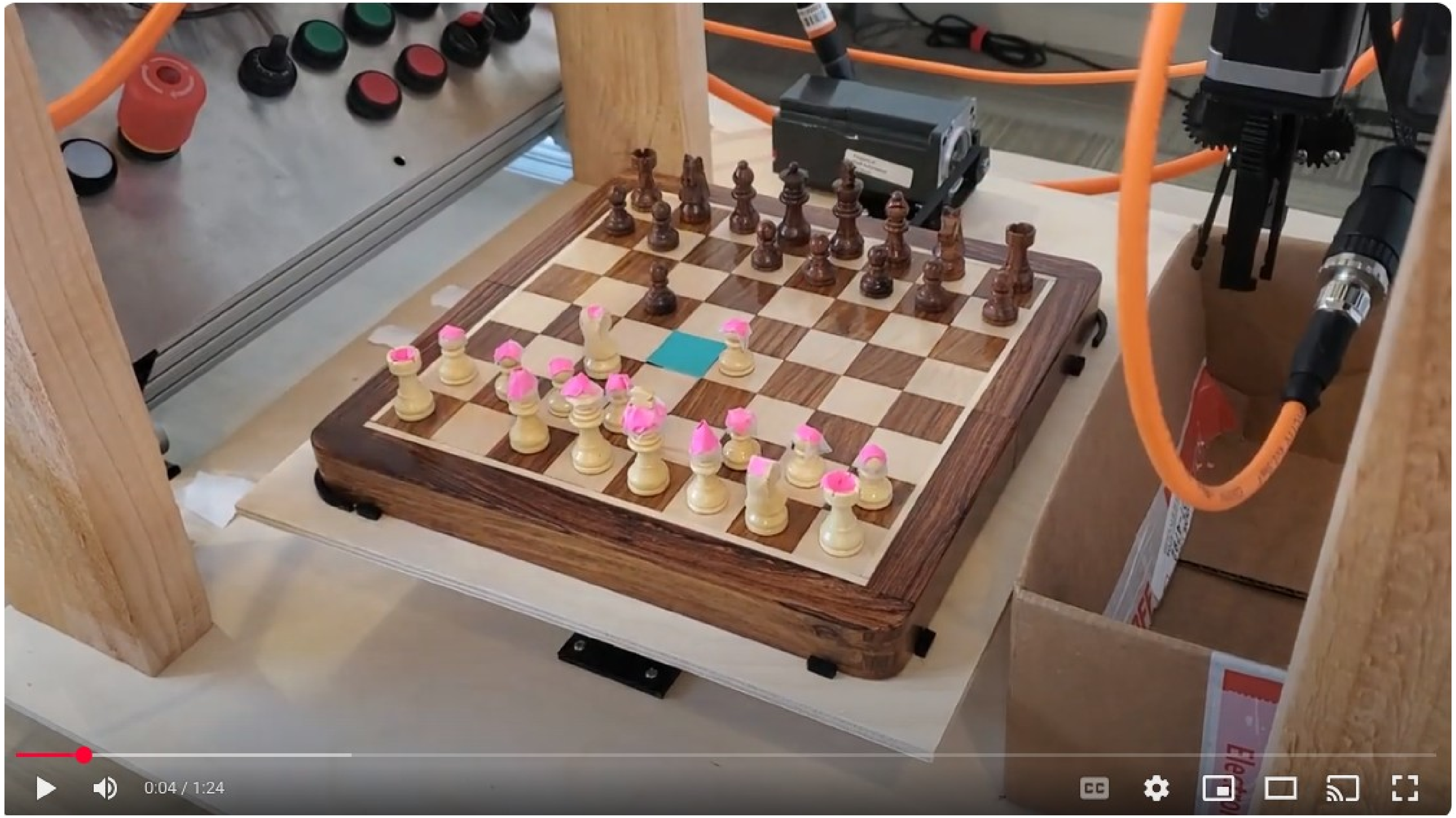
make me a script in python that reads an image and allows me to put my mouse over various pixels and return their hsv value

Here's a simple Python script using **OpenCV** that will:

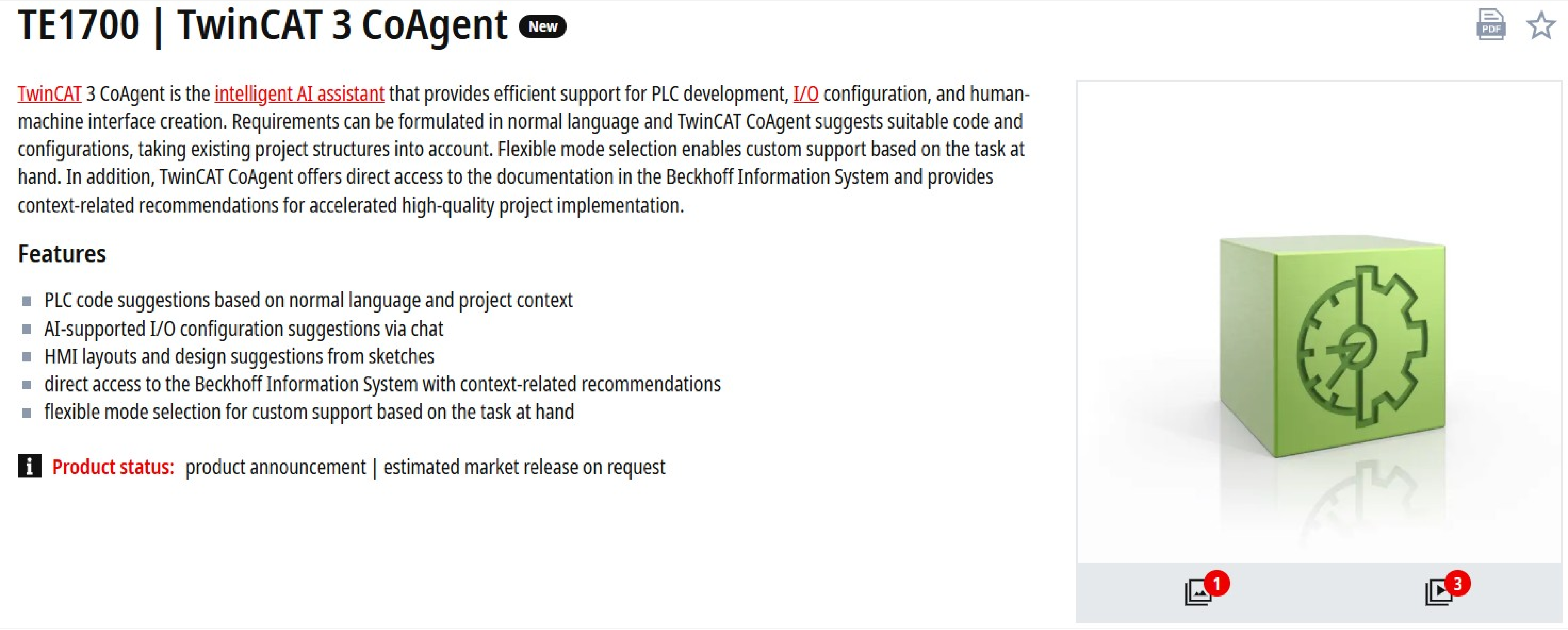
1. Load an image.
2. Display it in a window.
3. Let you hover your mouse over pixels.
4. Print the HSV (Hue, Saturation, Value) value at the cursor position in real time.

DEMO

https://www.youtube.com/watch?v=P32ndpw5Wa0



How is Beckhoff Adopting this Technology?



TE1700 | TwinCAT 3 CoAgent New

TwinCAT 3 CoAgent is the intelligent AI assistant that provides efficient support for PLC development, I/O configuration, and human-machine interface creation. Requirements can be formulated in normal language and TwinCAT CoAgent suggests suitable code and configurations, taking existing project structures into account. Flexible mode selection enables custom support based on the task at hand. In addition, TwinCAT CoAgent offers direct access to the documentation in the Beckhoff Information System and provides context-related recommendations for accelerated high-quality project implementation.

Features

- PLC code suggestions based on normal language and project context
- AI-supported I/O configuration suggestions via chat
- HMI layouts and design suggestions from sketches
- direct access to the Beckhoff Information System with context-related recommendations
- flexible mode selection for custom support based on the task at hand

Product status: product announcement | estimated market release on request

Thank you!