esportsLABgg League Vision Challenge
technical description of the task

1 Introduction

We are proud to announce **League Vision Challenge!**. EsportsLAB is in the stage of dynamic growth, and we are in search of motivated people that will help us develop a **next-generation AI powered** tool dedicated to better development of skills in playing computer games thanks to a data-driven approach to training methodology and creating automated recommendations.

Now, we like to announce **our programming challenge**, we propose a simple computer vision task related to **League Of Legends** MOBA computer game (LoL). Our challenge is not limited to LoL players, just knowledge of Python programming and basic computer vision is enough. We state it as an open problem, we do not enforce any particular approach, and leave it to your creativity. Authors of the best solutions will be awarded **Razer pro e-gaming equipment**, and will be invited to work with us on the next-gen computer vision technology during **generously paid internships** at our startup. The internship will be remote, and the duration of up to three months, with possible extension.

We will award three best solutions with

1. DeathAdder V2 Mouse + Kraken X headphones + Huntsman keyboard,
2. DeathAdder V2 Mouse + Huntsman keyboard,
3. Huntsman keyboard.

To be eligible for the reward you must be available and have an interest in participating in paid internships at esportsLABgg.

2 Task Details

Your main task is to write a script, which takes as input screens taken from League of Legends game replays, and outputs the exact count of the champion health bars that are visible on the screen.

Your solution should be a python script taking as **input a .jpg file game screen (name of the file is the command line parameter)** and prints out a single number – **the exact count of all visible health bars that belong to champions, not counting other visible health bars of other monsters and structures**. See the attached zip file with example input screens and the computed output.

2.1 Technology

The task should be implemented in **Python programming language**. We are not restricting usage of any specialized python libraries, you are free to use **any available external library** you wish.

Your solution must be submitted before the deadline (Aug 28, 11:59 PM CET) by email sent to lolchallenge@esportslab.gg.

2.2 Restrictions

It is forbidden to copy and paste any part of other participant solutions. Any violation of this rule and significant overlap of several solution scripts will result in the DQ of all submissions that are significantly similar or overlapping.

2.3 Problem Statement

INPUT Your solution takes as input a LoL ingame screen saved in .jpg file (its file-name is provided as a command line parameter).

OUTPUT Number $0 \leq n \leq 10$ of visible champion health bars.

Your goal is to write a computer vision python script that will count the health bars visible on the screen that are belonging to champions.

We will evaluate your solution against a varied choice of screens coming from a real game replay. Your solution needs to detect obscured health bars, and take into account several corner cases that occur in a realistic game setting. We mean precisely

- the health of a champion may be depleted, so only part of the color bar is visible,
- a health bar is hidden behind another health bar(s), behind some in-game text or some other elements of the game interface,
- health bars have two colors corresponding to two teams, and there may be parts of another color resulting from special spells,
- do not count health bars of other game objects and visible monsters.

We illustrate all of the corner cases in Fig. 2.

2.4 Example

On the example input presented in Fig. 1 there are five health-bars visible and hence your script should return

```
5
```

where 5 means that there are five champion health-bars visible on the screen.

You will find more example inputs and outputs in the accompanying zip file.

2.5 Evaluation Criteria

We will test your submission against ten other example screens taken in the same game. To evaluate the submitted solutions and select the winners we will use the following criteria

1. accurate identification of the number of health-bars visible on the screen in the test data,
2. code quality and its cleanliness.

2.6 Submission

Your solution python script shall be submitted before the deadline (Aug 28, 11:59 PM CET) by email sent to lolchallenge@esportslab.gg.

2.7 Further questions

Any questions related to the task you can send to the email address lolchallenge@esportslab.gg, or through our twitter account @esportsLABgg.

Good luck!!!

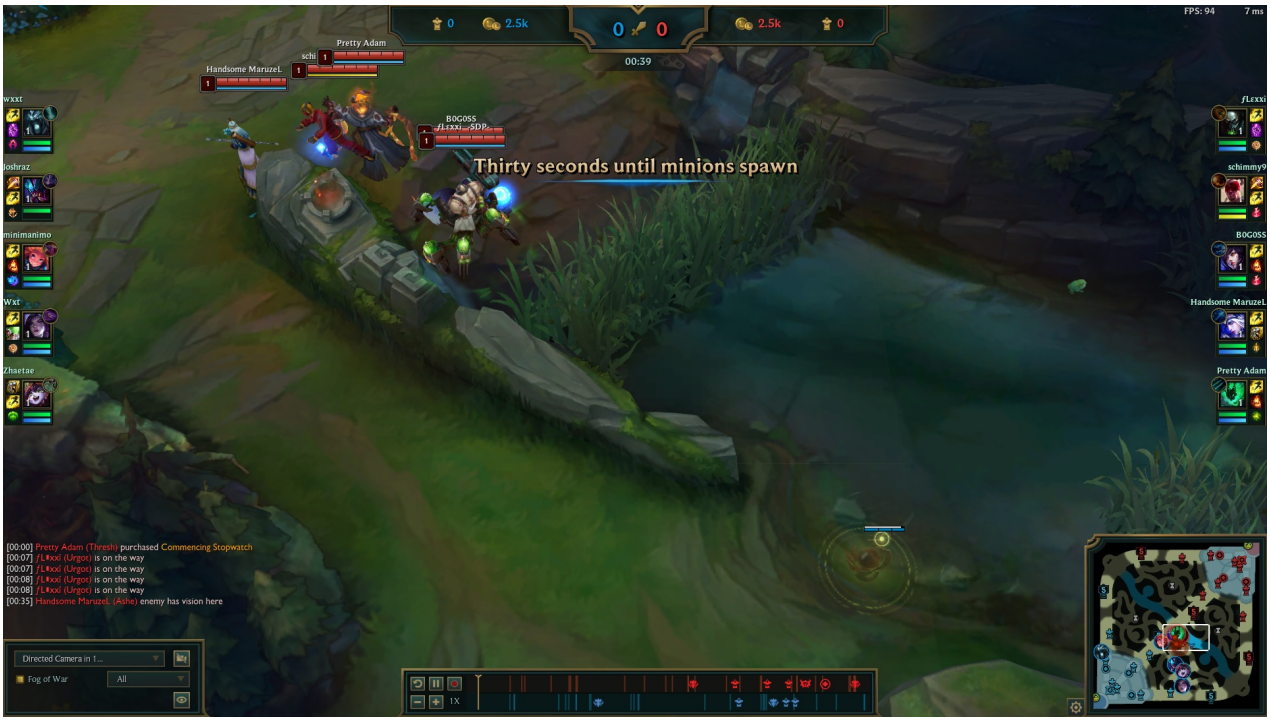
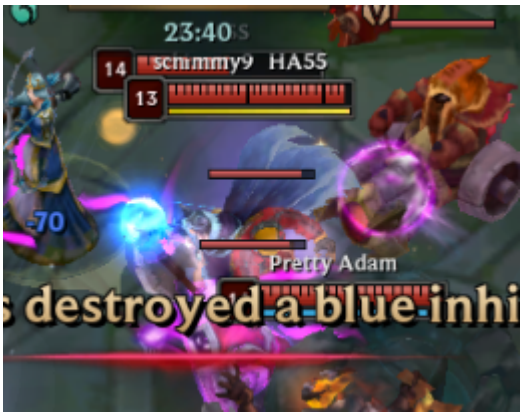


Figure 1: An example static screen from a League of Legends game replay, also example input that is provided to your solution script.

(a) Two champion health bars visible, one of them covered by an in game message, also visible one structure health bar (not to be counted).

(b) Two champion health bars visible, one of them is hidden behind another one.

(c) Three champion health bars visible. One is red color and two are blue color. All of them are partially drained. Additionally, one of the visible blue health bars has a red part resulting from a spell being casted at the champion.



(d) Three champion health bars visible. One of them is hidden behind an in game message, one is hidden behind an other health bar and a champion name.

(e) Two champion health bar visible. Part of one of them is hidden behind a visible game interface (a champion badge).

Figure 2: Subscreens taken from example game screens illustrating various corner cases for the health bar detection challenge task.