University of New Hampshire InterOperability Laboratory

University of New Hampshire

UTILIZING PROMINENT IMAGE COLORS TO CREATE SLIDESHOW THEMES

A powerful marketing and design tool to help anyone meet their presentation goals.

Authors

Iris Baird
Kyle Stewart

Affiliations

We'd like to thank the **Innovation Scholars** program here at UNH for a great opportunity to learn, and work hands on with developing an application. Thank you also to the **IOL** here on campus, and our mentors, Kyle Ouellette and Dean Sullivan.

Introduction

Our project aims to make creating aesthetic visuals easier. As students, we often create presentations to showcase research, inform and teach others, and organize ideas into a cohesive, understandable set of slides. For some, this comes easy, picking designs, colors, and formats, but for others, knowing where to start can be a challenge.

That is where our application comes in. The idea was that slides commonly require images of the subject and it would be easy to match those images to the actual presentation theme. That's where our application comes in, it allows you to insert any image, and using Amazon Web Services (AWS) Rekognition, it outputs the most prominent colors in hex codes. With these codes we insert them into a Google Slides API, that allows us to edit a set slide theme from a program! This creates visually appealing slides, ready for presenting!

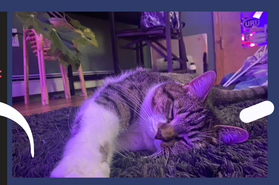
Methodology: Creating a Palette

The first step to creating this application was the programming, and utilization of the two APIs. We started with getting access to both, needing client secrets, and tokens for both to function properly.

AWS Rekognition:

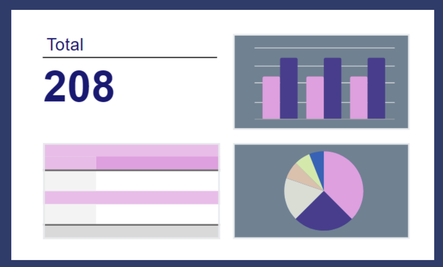
To start this we made an HTTP request to the API to get hex codes of the prominent colors of the inserted image.

```
    print(str(response))
    if "ImageProperties" in str(response):
        print("Colors detected:")
        for color in response["ImageProperties"]["DominantColors"]:
            print(color)

    return len(response['Labels'])

def main():
    photo = 'photoofmycat.jpg'
    label_count = detect_labels_local_file(photo)
    print("Labels detected: " + str(label_count))
```



```
ected labels in photoofmycat.jpg
ls': [], 'LabelModelVersion': '3.0', 'ImageProperties': {'Quality': {'Brightness'
Colors detected:
{'Red': 0, 'Blue': 0, 'Green': 0, 'HexCode': '#000000', 'CSSColor': 'black', 'Simplifie
{'Red': 72, 'Blue': 139, 'Green': 61, 'HexCode': '#483d8b', 'CSSColor': 'darkslateblue'
{'Red': 25, 'Blue': 112, 'Green': 25, 'HexCode': '#191970', 'CSSColor': 'midnightblue',
{'Red': 112, 'Blue': 144, 'Green': 128, 'HexCode': '#708090', 'CSSColor': 'slategrey',
{'Red': 221, 'Blue': 221, 'Green': 160, 'HexCode': '#dda0dd', 'CSSColor': 'plum', 'Simp
```

In this test of our program, we inserted a photo of a cat. The API detected the most prominent colors, creating a lovely color palette to input into the slide!



Methodology: Google Slides API

To create a theme from the prominent colors detected from the AWS API, we used the Google Slides API. This allows us to edit a theme from a program, which enables us to connect the two APIs to work as one.

- First, we created an app in the Google cloud workspace, then got access to the Google Slides API.
- This allows a user to login through Gmail, and allow the app permissions to their presentation.
- From this they can insert their image and the program will do the rest!
- The Google Slides API can do anything from create new slides, change themes, and even edit the content of the slide!

Project Evolution

When deciding on a project idea we knew we wanted to do something with colors. Our first idea was an application that could track website traffic and determine what popular website layouts and color palettes got the most. We wanted to use this to help web designers develop their site in a way that would get them noticed, and ultimately have the most success possible.

Unfortunately, there was no way to get the traffic information without asking for permissions from each site, which would be a hard and time-consuming task.

So we put our brains together and found a way to help anyone with their design needs, especially a way in that could help students, and schools, like us! Our project now is achievable, and incredibly helpful. We believe that with more time and work applied we could create something that can help anyone, from students to web designers, maybe even physical artists, or search engines.

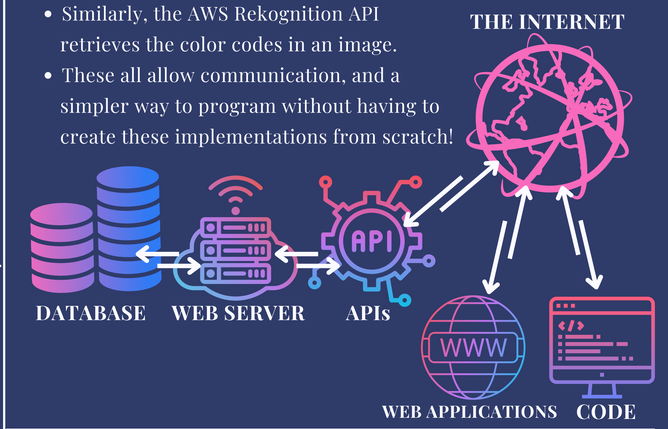
Image References: IOL Logo: from UNH Interoperability Lab. Python logo: Google Images. Amazon Rekognition Logo: pngkit.com. Google API logo: Google Images. All other elements from Canva.

What Are APIs?

Application Programming Interface

APIs are a way for two or more computer programs to communicate, and work together. Think of it like asking for advice from an expert. You have a plan or idea in mind, but need an outside resource to gather information, or perform a specific task for your program or application to work! That is what APIs do!

- In our program, we make a **GET** request to the Google Slides API to retrieve a presentation we can edit! It allows us to dissect and edit certain aspects of the slides from our application!
- Similarly, the AWS Rekognition API retrieves the color codes in an image.
- These all allow communication, and a simpler way to program without having to create these implementations from scratch!



Conclusion & Next Steps

In conclusion, we believe that our application can help lessen the stress of creating visually appealing presentations. Using the AWS Rekognition API and the Google Slides API, our application can detect prominent colors in images, extract the hex codes of the colors, and apply them to a presentation theme.

Our next steps would be to make our program more user-friendly, like a website or an app accessible to anyone. We would include new features, like formats and maybe extend to web design!