Project proposal

Guitar chord recognition

* We are going to investigate the problem of detecting guitar chords/notes from videos of people playing.
* The problem is very interesting because many people use youtube videos to learn new songs and to practice, but in order to understand what is the exact chord, they have to rewatch the same video many times. Our system allows to detect and recognize the finger patterns of a player in a video and generate the corresponding chord.
* We will examine many papers that already approached the same problem, such as:

<https://dl.acm.org/doi/pdf/10.1145/3290420.3290422?casa_token=Fq5IINLr_SAAAAAA:xxcOUjad7qwP7LkNrFiBNWCFVGTGkWdsSxyWEGY5zB33vGm1BvIlgIkp5CXMHR4FgHiE4n8MlVEo>

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fdiposit.ub.edu%2Fdspace%2Fbitstream%2F2445%2F171413%2F1%2FMITJANS%2520COMA%2520ALBERT_1338767_assignsubmission_file_TFG-Mitjans-Coma-Albert.pdf&psig=AOvVaw2zS2XkoBIKiYYSNmoSXZxy&ust=1715791283673000&source=images&cd=vfe&opi=89978449&ved=0CAcQrpoMahcKEwioxL3hyo2GAxUAAAAAHQAAAAAQBA>

<https://shawnbzhang.github.io/assets/PDFs/CS_230_Report.pdf>

* We intend to create our own dataset. We have two options:

1. Videotape one of us playing the basic chords, split the videos into frames, crop the frames to only get the hands and annotate the frames. Then we split the frames into training, validation and testing
2. We take videos of guitar players on youtube, use a chord-sound-identificator (like <https://github.com/ohollo/chord-extractor?tab=readme-ov-file> ) to get the correct chord, then we split the videos and annotate the frames and divide them into training, validation and testing

* We intend to study the literature to understand what are the implementations that give the most accurate results on the same kind of problems. Then, we want to train the best classifier on our dataset and try to obtain the best results by modifying some layers/parameters.
* The results will be evaluated in terms of accuracy of the chord predictions with respect to the ground truths. The metrics used will be the ones used for Classification: maximum accuracy, precision, recall. We can compare our results against the ones obtained in the cited papers by using CNNs (GoogleNet and ResNet) or YOLO.

PIPELINE:

1. Classical image-processing operators: we will use filters, such as Gaussian, Median, or Sobel filters to enhance edges or remove noise from the video frames, or Morphological operators to manipulate the shapes and structures within the image, which can help in segmenting the fingers and strings of the guitar.
2. Geometric-based algorithm or component: since the guitar player's fingers might be captured from different angles, we can use Homography transformation or Camera calibration techniques to correct the perspective distortion.
3. Retrieval algorithm: after extracting the features, we will retrieve the closest reference pattern of fingers position and known chord from our dataset, using algorithms such as Template matching or Feature-based matching like SIFT.
4. Deep learning-based component: we will train a CNN to learn to recognize the spatial arrangements of fingers/ fingers patterns for each chord.

Photorealism enhancement (image-to-image translation)

* We are going to investigate the problem of making rendered or synthetic images more realistic. The problem of achieving realistic images is crucial for various applications, such as gaming, where realistic visuals enhance user immersion and engagement, virtual and augmented reality, simulation, digital art and photoraphy, and also scientific or medical visualization, where having realistic images aids in interpretation and communication of complex data.
* This problem has already been highly approached during the past, so we will examine many papers relative to image enhancing and translation such as:

<https://paperswithcode.com/paper/enhancing-photorealism-enhancement>

<https://paperswithcode.com/task/synthetic-to-real-translation>

<https://paperswithcode.com/task/image-to-image-translation>

<https://paperswithcode.com/paper/art2real-unfolding-the-reality-of-artworks>

<https://openaccess.thecvf.com/content_ECCV_2018/papers/Chuanxia_Zheng_T2Net_Synthetic-to-Realistic_Translation_ECCV_2018_paper.pdf>

* As regards the dataset, we will evaluate three options:

1. We can create our own dataset, selecting real photos from Internet and create their corresponding rendered version by applying filters, such as blurring, color adjustements, noise and texture filters, artistic filters and geometric transformations, such as perspective distortion, rotation, …
2. We can use the LAION-400-MILLION dataset, that contains 400 million CLIP-filtered image-text pairs. We would select a subset of real images from LAION and pair them with the relative filtered images throgh the CLIP filtering. We would then preprocess them, using filters and geometrical transformations.
3. We can select synthetic images from a dataset like GTAV-Cityscapes, and use a CNN like ResNet to find and retrieve the similar real images from another dataset, such as Cytiscapes, COCO or OpenImages. We would then do the preprocessing in a similar way to option 2.

* We intend to study the literature to understand what are the methods that give the best and most realt-time results on this kind of task. In particular, we have to decide if it’s better to use a GAN, such as SimGAN or CycleGAN, or use a DIFFUSION model. In any case, the idea is to use a pretrained network, given the limited amount of computational power at our disposal, finetune it on our dataset and modify some layers/parameters in order to obtain the most real-like images.
* The results will be evaluated in terms of FID and CLIP, to evaluate the quality of the generated images, and in terms of Recall, for what regards the retrieval of real images from the dataset, similar to the generated ones. We can compare our results against the ones obtained in the cited papers by using other types of networks (es another type of GAN).

PIPELINE:

If we choose to create our own dataset, we will use **Classical image-processing operators,** such as Bilateral filter, Posterization filter, Gaussian blur, and **Geometric-based algorithm**, such as Rotation and Perspective Transformation to create the corresponent rendered images to real ones. While if we choose to use an existing dataset, the filters and transformations will be applied to preprocess the images.

The **Retrieval algorithm,** such as Template matching or Feature-based matching like SIFT, will be used to find the real images from the dataset similar to the images generated from our network. This step is essential in the evaluating stage. Moreover, if we choose option 3 in the dataset choice, a Retrieval algorithm will be used to find the correspondent real images from the synthetic ones of the GTAV dataset.

Lastly, as **Deep learning-based component** we will implement a generative network, such as GAN or DIFFUSION, and finetune it by training the last layers.