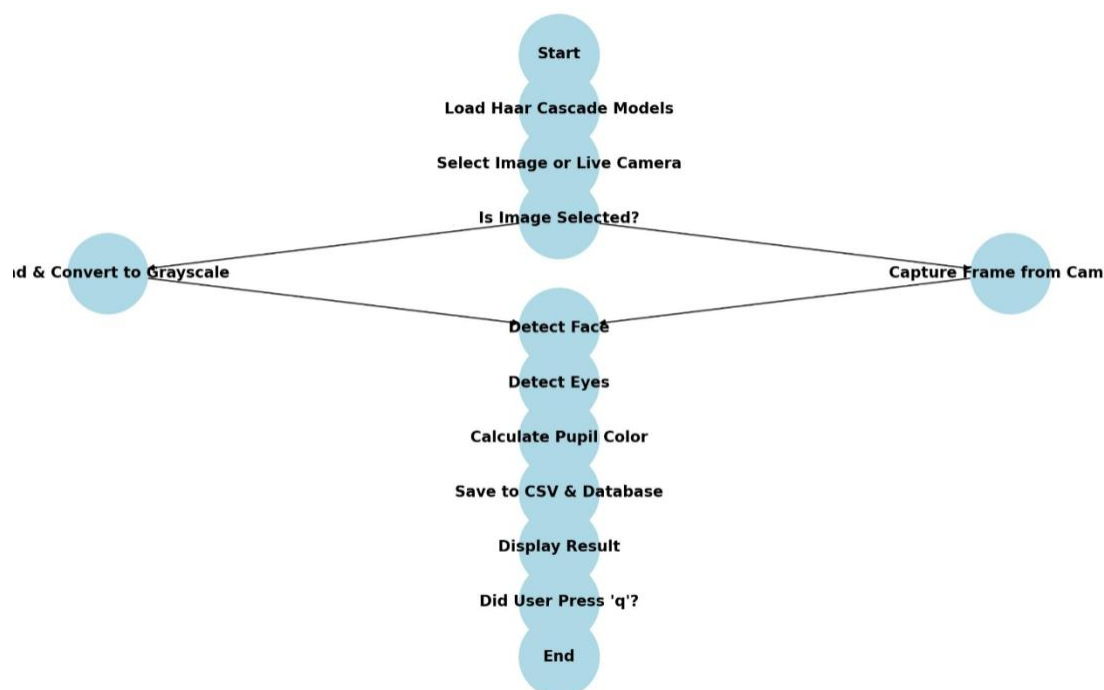


# الگوریتم تشخیص چهره و رنگ مردمک چشم



## 1 بارگذاری مدل‌های تشخیص

مدل‌های Haar Cascade برای تشخیص چهره و چشم از OpenCV بارگذاری می‌شوند.

## 2 انتخاب تصویر یا استفاده از دوربین زنده

کاربر می‌تواند یک تصویر را انتخاب کند یا از دوربین زنده استفاده کند.

اگر تصویر انتخاب شد، آن را بارگذاری می‌کنیم.

اگر دوربین فعال شد، فریم‌ها را به صورت زنده دریافت می‌کنند

### 3 پردازش تصویر

تبدیل تصویر به سطح خاکستری ( Grayscale) برای بهبود تشخیص.

استفاده از مدل Haar Cascade برای تشخیص چهره در تصویر.  
در هر چهره، تشخیص چشم‌ها انجام می‌شود.

### 4 تشخیص رنگ مردمک چشم

ناحیه چشم جدا شده و کمی بلور ( Gaussian Blur) روی آن اعمال می‌شود.

تصویر چشم به مدل HSV تبدیل می‌شود.

مقدار میانگین رنگ HSV در ناحیه مردمک محاسبه می‌شود.

مقدار رنگ به‌صورت HSV روی تصویر نمایش داده می‌شود.

## 5 ذخیره اطلاعات

مقدار رنگ HSV در فایل CSV ذخیره می‌شود.

مقدار رنگ HSV در دیتابیس SQLite ذخیره می‌شود.

## 6 نمایش نتیجه

تصویر پردازش‌شده در GUI با Tkinter نمایش داده می‌شود.

در حالت زنده، تصویر پردازش‌شده روی صفحه نمایش داده می‌شود.

برای خروج از دوربین زنده، کاربر می‌تواند کلید  
"q" را فشار دهد.

نکات مهم در این الگوریتم:

✓ استفاده از Haar Cascade برای تشخیص  
چهره و چشم‌ها.

✓ اعمال Gaussian Blur برای کاهش نویز و  
بهبود دقت تشخیص رنگ.

✓ ذخیره اطلاعات رنگ مردمک در CSV و  
دیتابیس SQLite.

✓ رابط گرافیکی با Tkinter برای انتخاب تصویر  
و نمایش نتیجه.

✓ امکان تشخیص زنده از دوربین برای بررسی  
رنگ مردمک چشم در لحظه.