

Face_Detection_Sample.java

```
class Face_Detection_View extends View {
    private static final int MAX_FACES = 10;
    private static final String IMAGE_FN = "face.jpg";
    private Bitmap background_image;
    private FaceDetector.Face[] faces;
    private int face_count;

    // preallocate for onDraw(...)
    private PointF tmp_point = new PointF();
    private Paint tmp_paint = new Paint();

    public Face_Detection_View(Context context) {
        super(context);
        // Load an image from SD Card
        updateImage(Environment.getExternalStorageDirectory() + "/" + IMAGE_FN);
    }

    public void updateImage(String image_fn) {
        // Set internal configuration to RGB_565
        BitmapFactory.Options bitmap_options = new BitmapFactory.Options();
        bitmap_options.inPreferredConfig = Bitmap.Config.RGB_565;

        background_image = BitmapFactory.decodeFile(image_fn, bitmap_options);
        FaceDetector face_detector = new FaceDetector(
            background_image.getWidth(), background_image.getHeight(),
            MAX_FACES);

        faces = new FaceDetector.Face[MAX_FACES];
        // The bitmap must be in 565 format (for now).
        face_count = face_detector.findFaces(background_image, faces);
        Log.d("Face_Detection", "Face Count: " + String.valueOf(face_count));
    }

    public void onDraw(Canvas canvas) {
        canvas.drawBitmap(background_image, 0, 0, null);
        for (int i = 0; i < face_count; i++) {
            FaceDetector.Face face = faces[i];
            tmp_paint.setColor(Color.RED);
            tmp_paint.setAlpha(100);
            face.getMidPoint(tmp_point);
            canvas.drawCircle(tmp_point.x, tmp_point.y, face.eyesDistance(),
                tmp_paint);
        }
    }
}
```