



Multimédia

sous android



Plan

I.	VIBREUR	1
II.	CAPTURE PHOTO	1
II.1.	LANCER L'INTENT DE CAPTURE DE LA PHOTO	1
II.2.	RECUPERER LA PHOTO CAPTUREE	1
III.	CAPTURE VIDEO	2
III.1.	LANCER L'INTENT DE CAPTURE DE LA VIDEO	2
III.2.	RECUPERER LA VIDEO CAPTUREE	2
IV.	ML KIT	2
V.	EXEMPLE DE DETECTION DE VISAGE	2
VI.	EXEMPLE DE RECONNAISSANCE DES CODES-BARRES	4

Références :

- [1] <http://android-france.fr/2009/09/utiliser-en-developpement-le-service-vibreur-de-votre-telephone-android-dans-vos-applications/>
- [2] <https://developer.android.com/training/camera/photobasics.html>
- [3] <https://developer.android.com/training/camera/videobasics.html>
- [4] <https://developers.google.com/ml-kit?hl=fr>

I. Vibreur

```
<uses-permission android:name="android.permission.VIBRATE" />
Vibrator v;
v = (Vibrator) getSystemService(Context.VIBRATOR_SERVICE);
int duree=100; //en milliseconde
v.vibrate(duree);
```

II. Capture photo

II.1. Lancer l'intent de capture de la photo

```
<uses-feature android:name="android.hardware.camera"
              android:required="true" />
static final int REQUEST_IMAGE_CAPTURE = 1;

private void dispatchTakePictureIntent() {
    Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
        startActivityForResult(takePictureIntent, REQUEST_IMAGE_CAPTURE);
    }
}
```

II.2. Récupérer la photo capturée

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == REQUEST_IMAGE_CAPTURE && resultCode == RESULT_OK) {
        Bundle extras = data.getExtras();
        Bitmap imageBitmap = (Bitmap) extras.get("data");
        mImageView.setImageBitmap(imageBitmap);
    }
}
```

III. Capture Video

III.1. Lancer l'intent de capture de la vidéo

```
<uses-feature android:name="android.hardware.camera"
              android:required="true" />
static final int REQUEST_VIDEO_CAPTURE = 2;
private void dispatchTakeVideoIntent() {
    Intent takeVideoIntent = new Intent(MediaStore.ACTION_VIDEO_CAPTURE);
    if (takeVideoIntent.resolveActivity(getPackageManager()) != null) {
        startActivityForResult(takeVideoIntent, REQUEST_VIDEO_CAPTURE);
    }
}
```

III.2. Récupérer la vidéo capturée

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent intent) {
    if (requestCode == REQUEST_VIDEO_CAPTURE && resultCode == RESULT_OK) {
        Uri videoUri = intent.getData();
        mVideoView.setVideoURI(videoUri);
        mVideoView.start();
    }
}
```

IV. ML Kit

ML kit est le machine-learning pour les développeurs d'applications mobiles. ML Kit met l'expertise de Google en machine learning à la disposition des développeurs mobiles dans un package performant et facile à utiliser. Le traitement par ML Kit s'effectue sur l'appareil. Elle est ainsi rapide et permet d'utiliser des cas d'utilisation en temps réel, comme le traitement de l'entrée de l'appareil photo. Elle fonctionne également hors connexion, et peut être utilisée pour traiter les images et le texte qui doivent rester sur l'appareil [4].

V. Exemple de détection de visage

```
//StatVisage
public class MainActivity extends AppCompatActivity {
    private ImageView imgVisages;
    private Button btnAnalyser;
    private TextView tvNbV;
    private TextView tvNbVS;
    private TextView tvNbVI;
    private TextView tvDYM;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);          initGraphique();
    }
    private void initGraphique () {
        ...
        ajouterEcouteur();
    }
    private void ajouterEcouteur() {
        btnAnalyser.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                analyser();
            }
        });
    }
}
```

```

private void analyser() {
    Bitmap bitmap = ((BitmapDrawable) imgVisages.getDrawable()).getBitmap();
    InputImage image = InputImage.fromBitmap(bitmap, 0);
    FaceDetectorOptions options =
        new FaceDetectorOptions.Builder()

.setPerformanceMode(FaceDetectorOptions.PERFORMANCE_MODE_ACCURATE)
        .setLandmarkMode(FaceDetectorOptions.LANDMARK_MODE_ALL)

.setClassificationMode(FaceDetectorOptions.CLASSIFICATION_MODE_ALL)
        .setMinFaceSize(0.15f)
        .enableTracking()
        .build();
    FaceDetector detector = FaceDetection.getClient(options);
    Task<List<Face>> result =
        detector.process(image)
            .addOnSuccessListener(
                new OnSuccessListener<List<Face>>() {
                    @Override
                    public void onSuccess(List<Face> faces) {
                        afficherStat(faces);
                    }
                })
            .addOnFailureListener(
                new OnFailureListener() {
                    @Override
                    public void onFailure(@NonNull Exception e)
{
Toast.makeText(MainActivity.this, "Failure: "+e.getMessage(), Toast.LENGTH_SHORT).show();
                }
            });
}

private void afficherStat(List<Face> faces) {
    int nbV=faces.size();
    int nbVS=0;
    int nbVI=0;
    double dymMax=0;
    for (int i = 0; i < faces.size(); i++) {
        Face face = faces.get(i);
        if(face.getSmilingProbability()>0.5f)                nbVS++;
        if(face.getHeadEulerAngleZ()>20)                    nbVI++;
        double xLE = 0, yLE = 0, xRE = 0, yRE = 0;
        for (FaceLandmark l : face.getAllLandmarks()) {
            if (l.getLandmarkType() == FaceLandmark.LEFT_EYE) {
                xLE = l.getPosition().x;
                yLE = l.getPosition().y;
            }
            if (l.getLandmarkType() == FaceLandmark.RIGHT_EYE) {
                xRE = l.getPosition().x;
                yRE = l.getPosition().y;
            }
        }
        double dym = Math.sqrt(Math.pow(xLE - xRE, 2) + Math.pow(yLE - yRE, 2));
        if (dym>dymMax)
            dymMax=dym;
    }
    tvNbV.setText(nbV+"");
    tvNbVS.setText(nbVS+"");
    tvNbVI.setText(nbVI+"");
    tvDYM.setText(dymMax+"");
}
}

```

VI. Exemple de reconnaissance des codes-barres

```
//PrixProduit
public class MainActivity extends AppCompatActivity {
    private Button btnSuivant;    private Button btnAnalyser;    private ImageView
imgProduit;    private TextView tvPrix;
    private int[] tImage = new int[] {
R.drawable.eau_m1,R.drawable.eau_m2,R.drawable.eau_safia };
    private int indiceCourant;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        ...
    }
    private void initGraphique() {
        ...
        indiceCourant=0;        ajouterEcouteurs();
    }
    private void ajouterEcouteurs() {
        btnAnalyser.setOnClickListener(view -> analyser());
        btnSuivant.setOnClickListener(view -> suivant());
    }
    private void suivant() {
        indiceCourant = (indiceCourant + 1) % tImage.length;
        imgProduit.setImageResource(tImage[indiceCourant]);
        analyser();
    }
    private void analyser() {
        Bitmap bitmap = ((BitmapDrawable) imgProduit.getDrawable()).getBitmap();
        InputImage image = InputImage.fromBitmap(bitmap, 0);
        BarcodeScanner scanner = BarcodeScanning.getClient();
        Task<List<Barcode>> result = scanner.process(image)
            .addOnSuccessListener(new OnSuccessListener<List<Barcode>>() {
                @Override
                public void onSuccess(List<Barcode> barcodes) {
                    afficherPrix(barcodes);
                }
            })
            .addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(MainActivity.this, "Failure:"+e.getMessage(), Toast.LENGTH_SHORT).show();
                }
            });
    }
    private void afficherPrix(List<Barcode> barcodes) {
        if(barcodes.size()>0){
            if(barcodes.size()==1){
                Barcode bar = barcodes.get(0);
                tvPrix.setText(getPrix(bar.getRawValue()));
            }else
                Toast.makeText(this, "Plusieurs code à barre dans l'image!", Toast.LENGTH_SHORT).show();
            }else
                Toast.makeText(this, "Aucun code à bar dans l'image!", Toast.LENGTH_SHORT).show();
        }
        private String getPrix(String rawValue) {
            String prix="Le prix ";
            if(rawValue.equals("6191577600015"))
                prix+="de Eau Safia est 0.900 DT";
            else if(rawValue.equals("123456789012"))
                prix+="de Eau M1 est 6.700 DT";
            else if(rawValue.equals("3401312345624"))
                prix+="de Eau M2 est 4.300 DT";
            else    prix+="Inconnu";
            return prix;
        }
    }
}
```