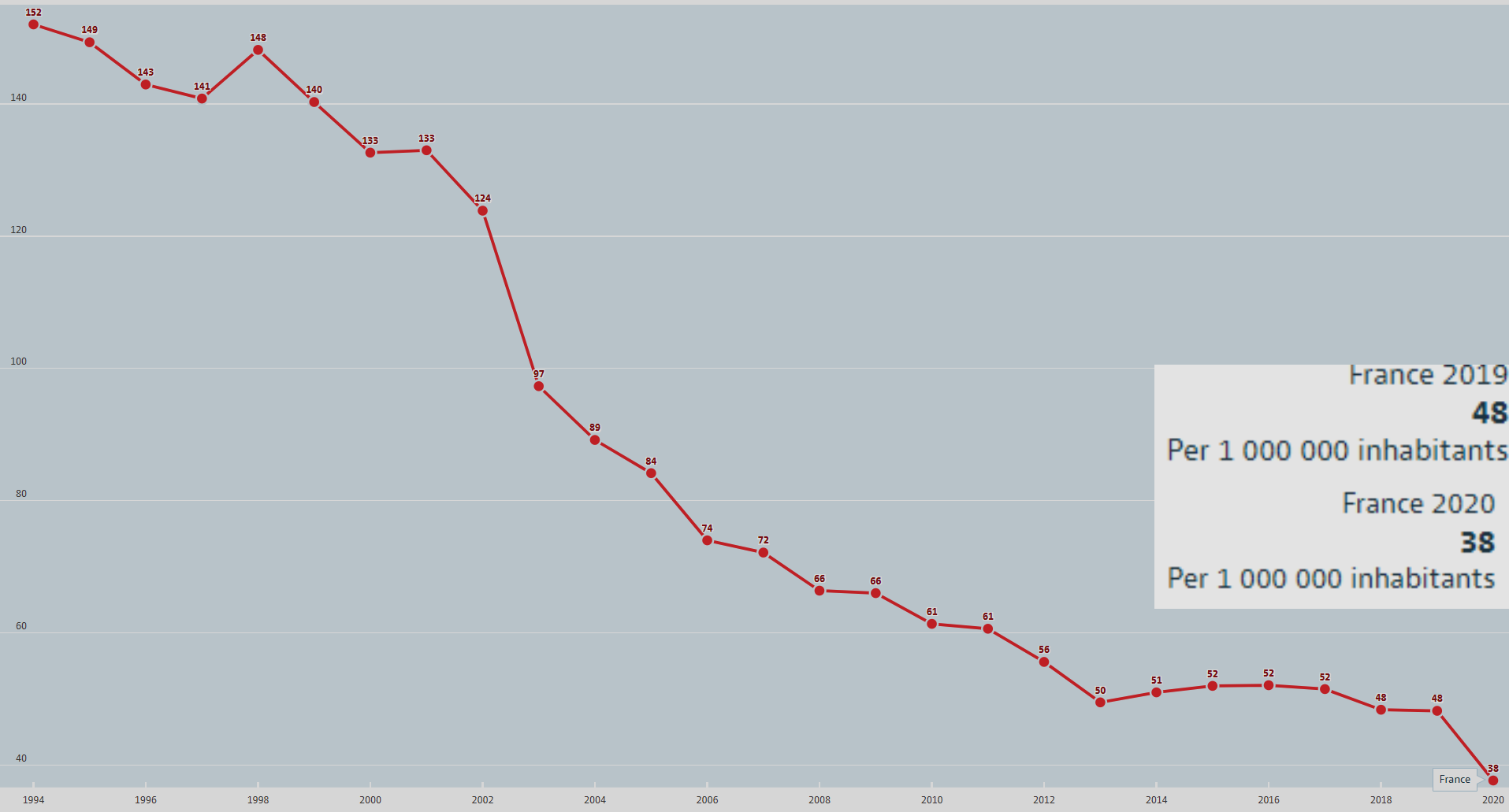


MAAF

+



Road accidents Deaths, Per 1 000 000 inhabitants, 1994 – 2020



Le lien entre les décès sur les routes et la fatigue



37% DES 153 DÉCÈS
SURVENUS SUR AUTOROUTE EN 2011
ÉTAIENT DUS À LA SOMNOLENCE

137.2 01:03 79

Problématique : Comment faire comprendre à une voiture que son conducteur est somnolent ?

Plan :

Introduction : Présentation des deux méthodes traitées

I : La méthode cardiaque

- 1° Le principe de la méthode
- 2° L'étude du filtre passe-bas d'ordre 2
- 3° Les acquisitions et leur analyse

II : La méthode informatique

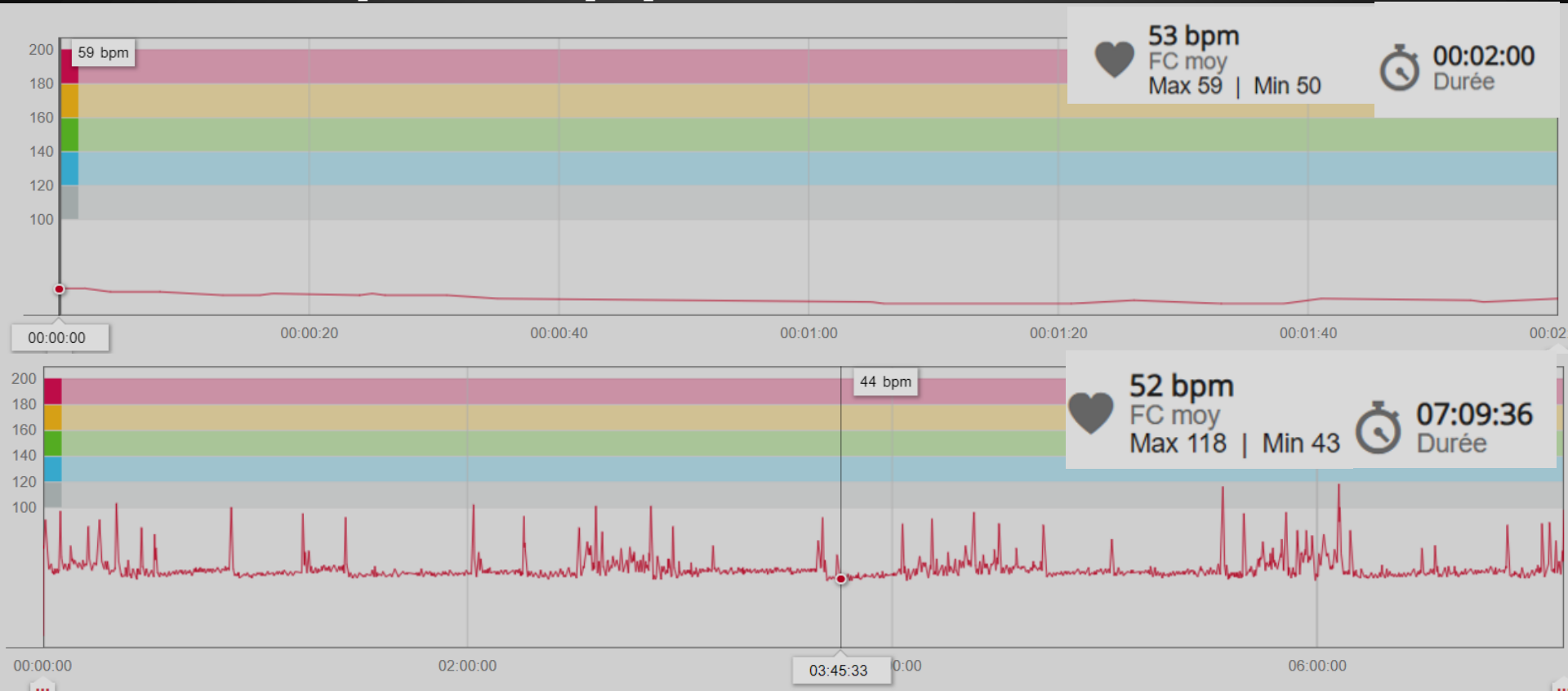
- 1° Le principe de la méthode
- 2° Les paramètres choisis
- 3° Les résultats de l'entraînement
- 4° Application avec une webcam

III : Les limites des méthodes

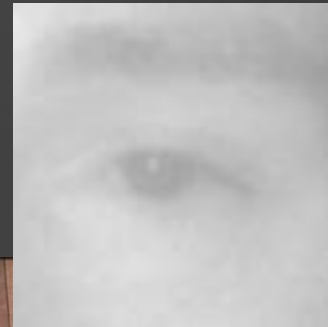
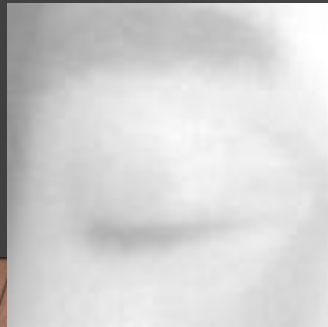
- 1° Les limites de la méthode cardiaque
- 2° Les limites de la méthode informatique

Deux méthodes pour détecter la fatigue

1° La baisse de la fréquence cardiaque pendant le



2° Le fermement des yeux



I 1: Principe de la méthode cardiaque



1° Acquérir le signal cardiaque
au niveau du cœur

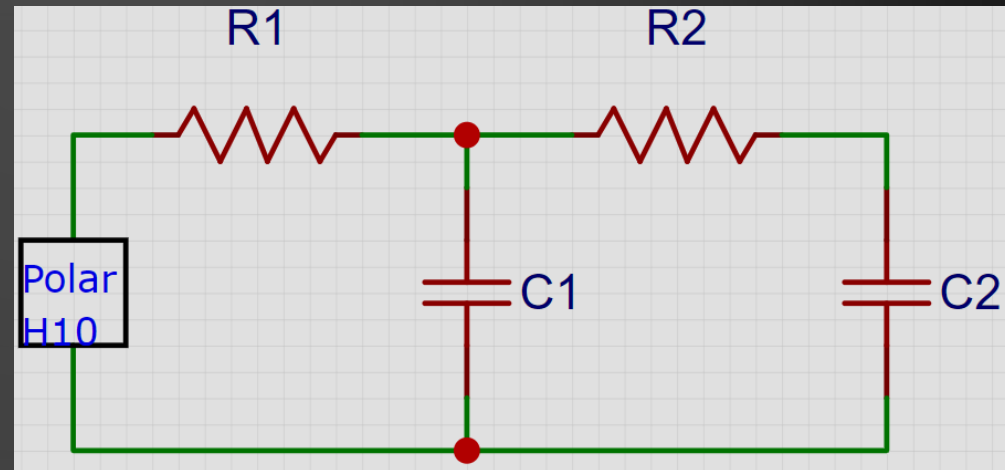
- POLAR H10
- ECGLogger
- Latis-Pro.

Caractéristiques des signaux:

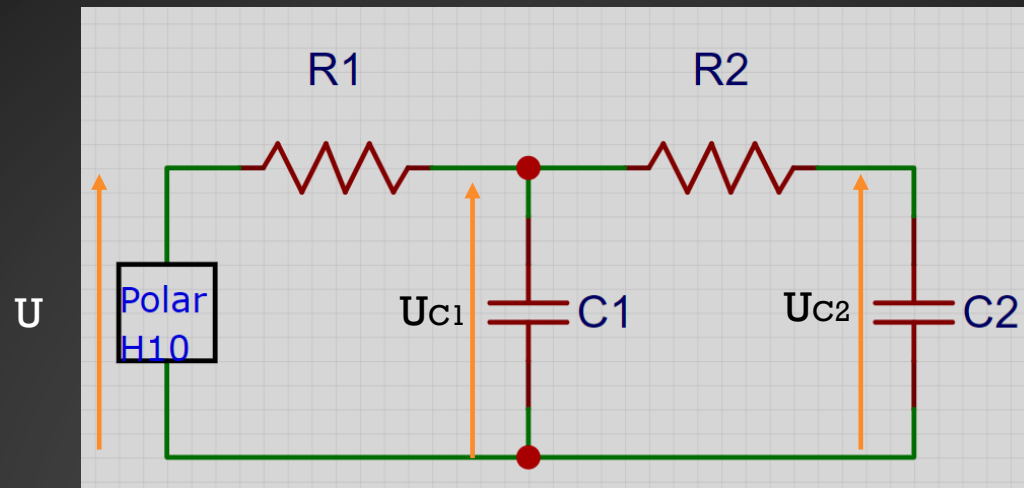
- $-\Delta t = 10 \text{ s}$
- $-T_e = 7 \text{ ms}$

2° Appliquer un filtre passe-bas d'ordre 2 dont
la fréquence de coupure est 4 Hz.

3° Grâce à une analyse de Fourier faite par
Latis-Pro, déterminer la fréquence cardiaque.



I 2: Étude du filtre passe-bas d'ordre 2



$$R1=R2=R$$

$$C1=C2=C$$

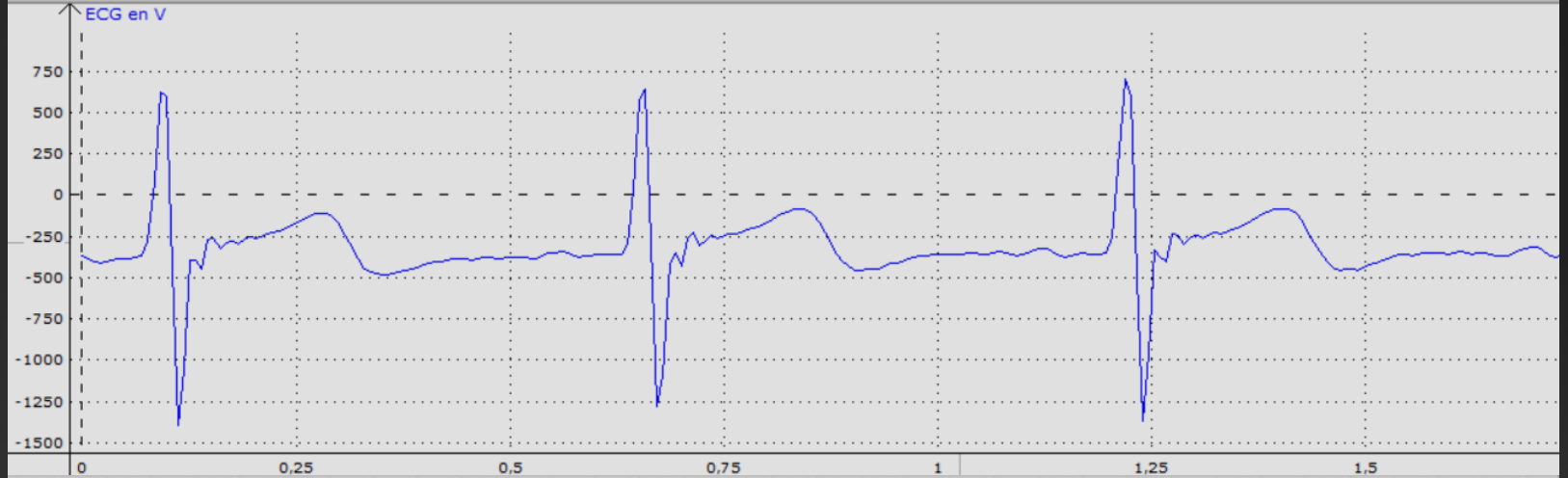
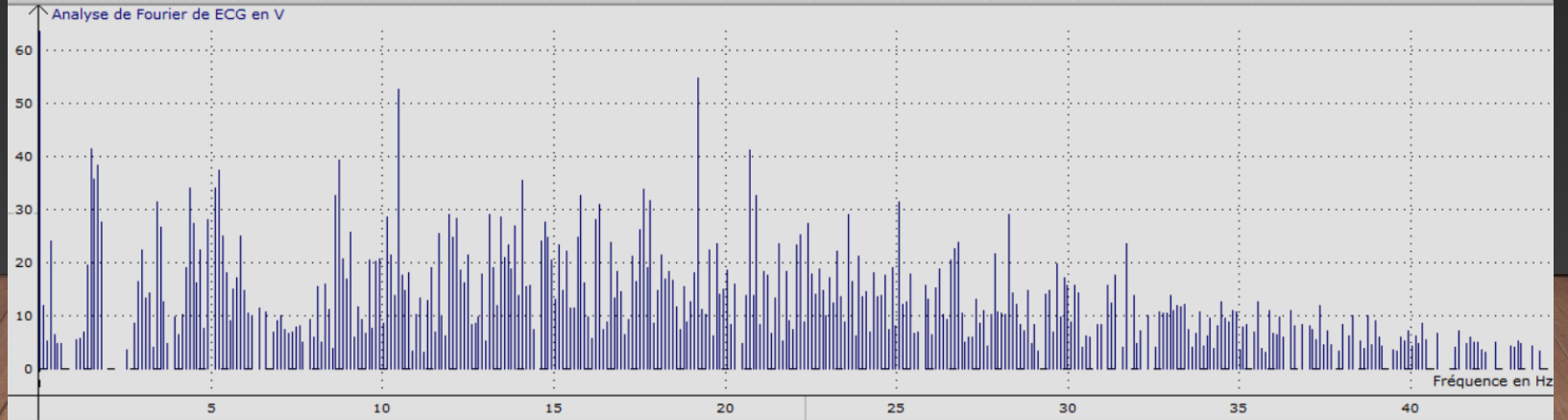
$$U_{C2}'' + \frac{3}{RC} * U_{C2}' + \frac{1}{RC^2} * U_{C2} = \frac{1}{RC^2} * U$$

Avec $f_{coupure} = 4Hz$ on trouve $RC = 2,56 * 10^{-2}s$

I 3 a: Signal directement sur la peau

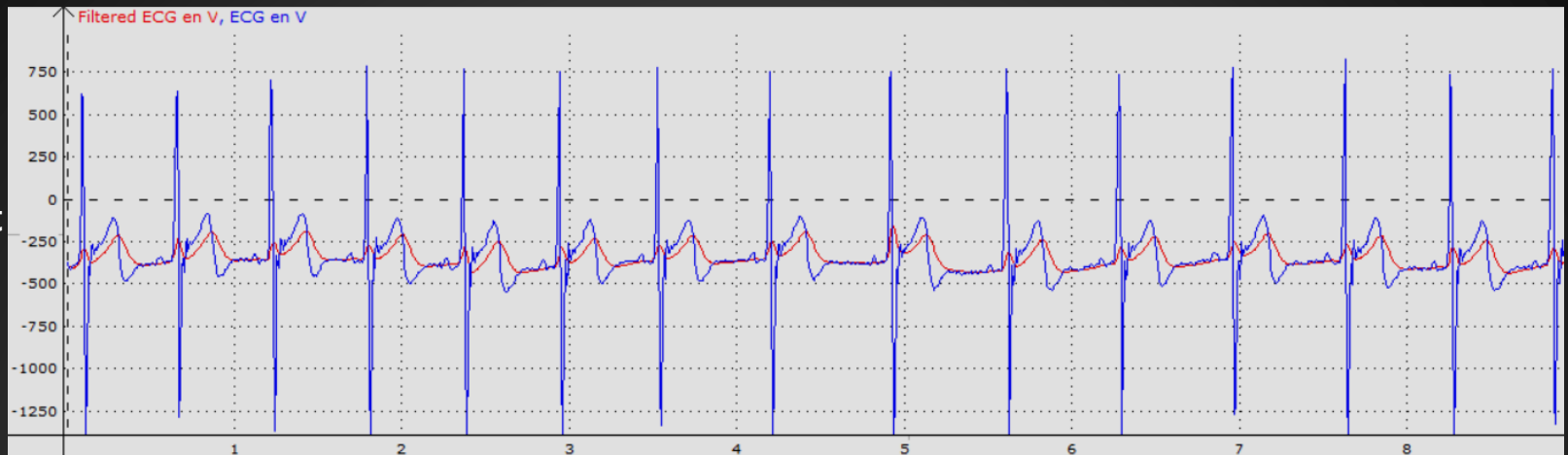
Signal
obtenu

Zoom

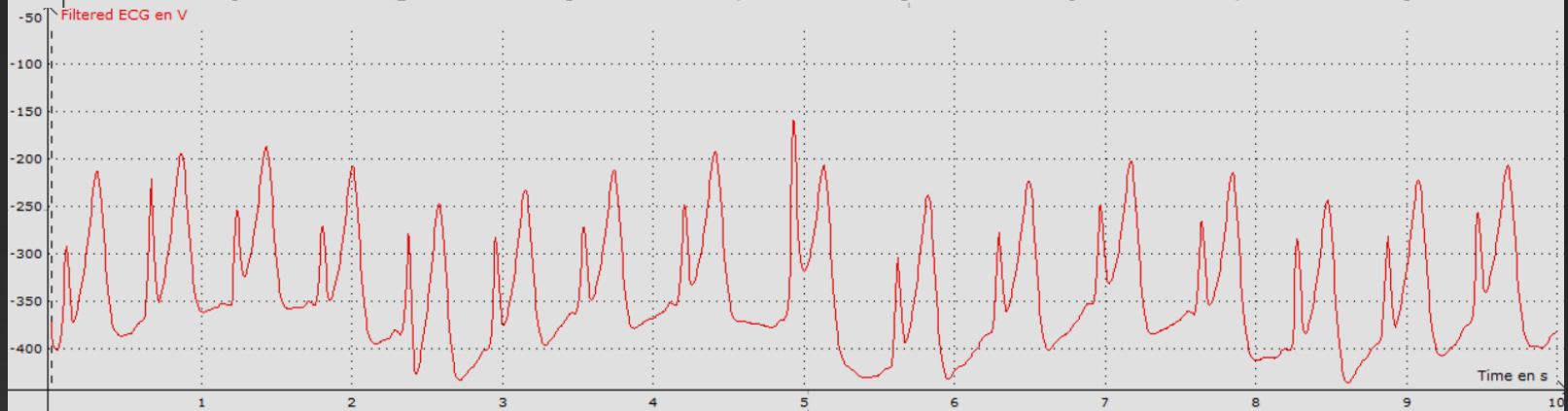
Analyse
de
Fourier

I 3 a: Signal directement sur la peau

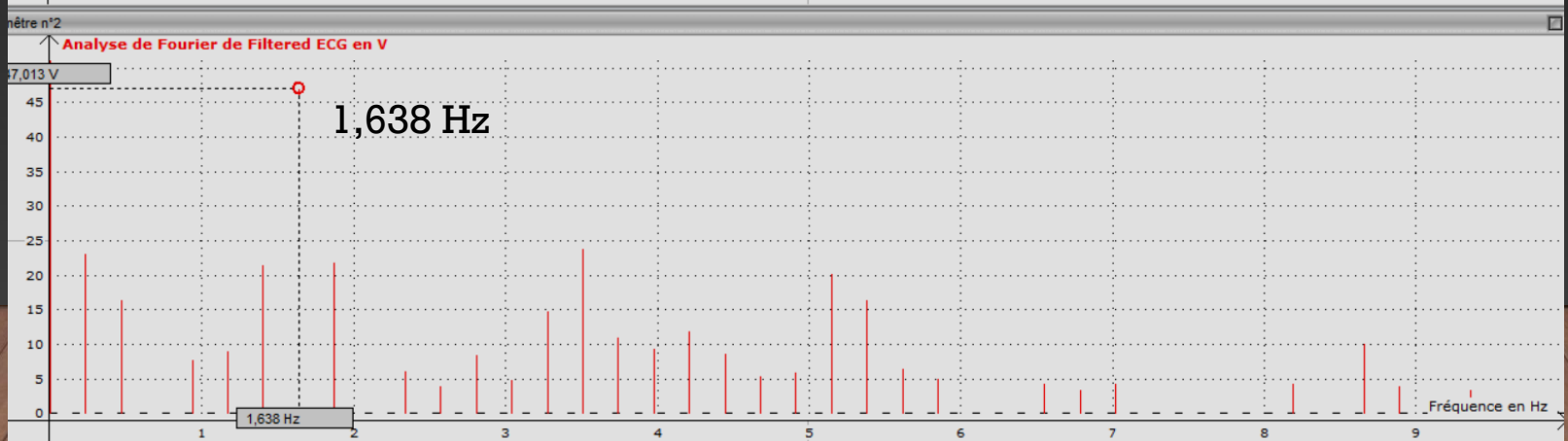
Signal
obtenu et
signal
filtré



Signal
filtré



Analyse
de
Fourier
signal
filtré



Par-dessus un T-Shirt: 1,823 Hz



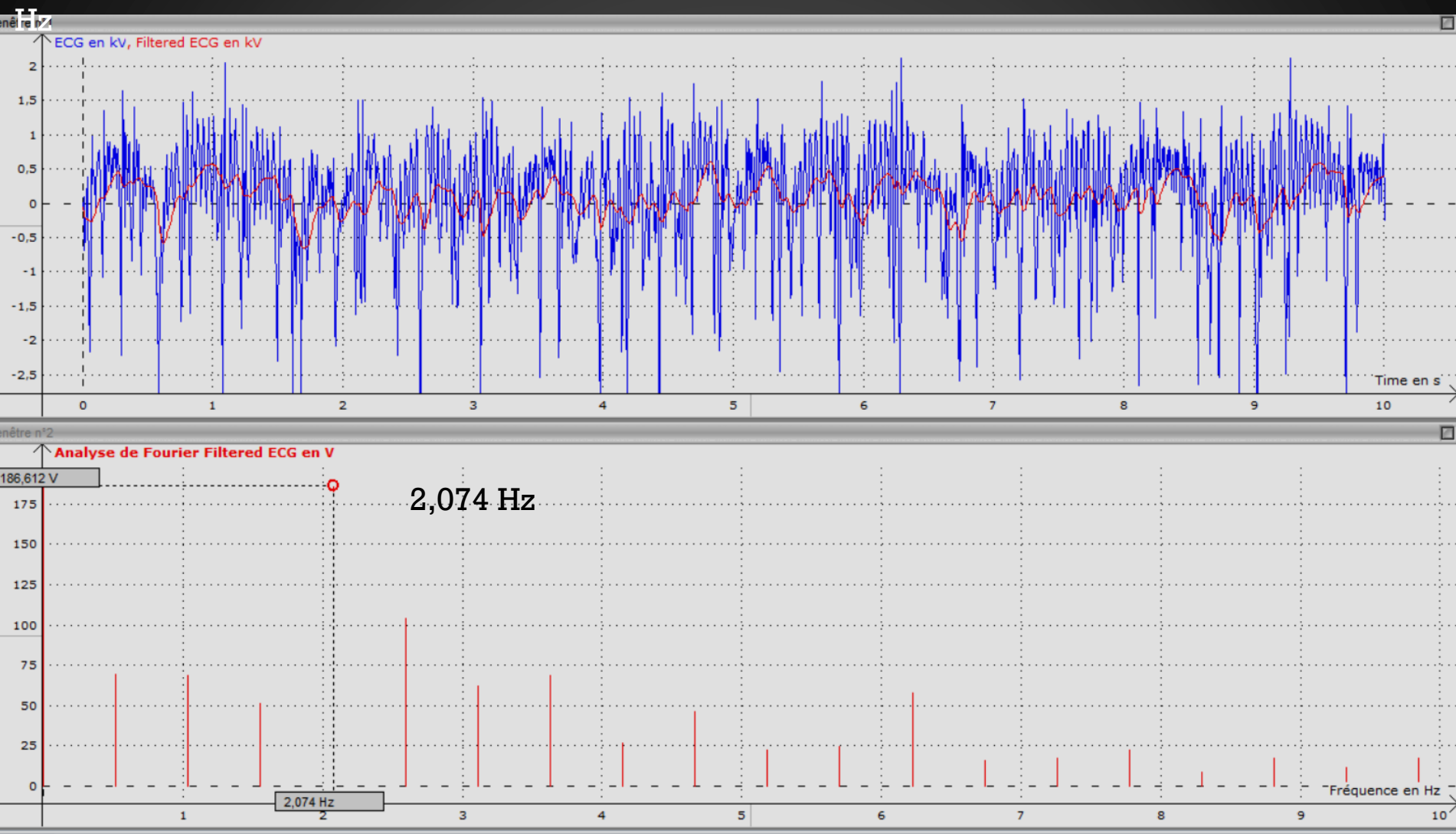
I 3 b: Signaux par-dessus plusieurs couches vestimentaires

Par-dessus deux T-Shirt: 1,47 Hz

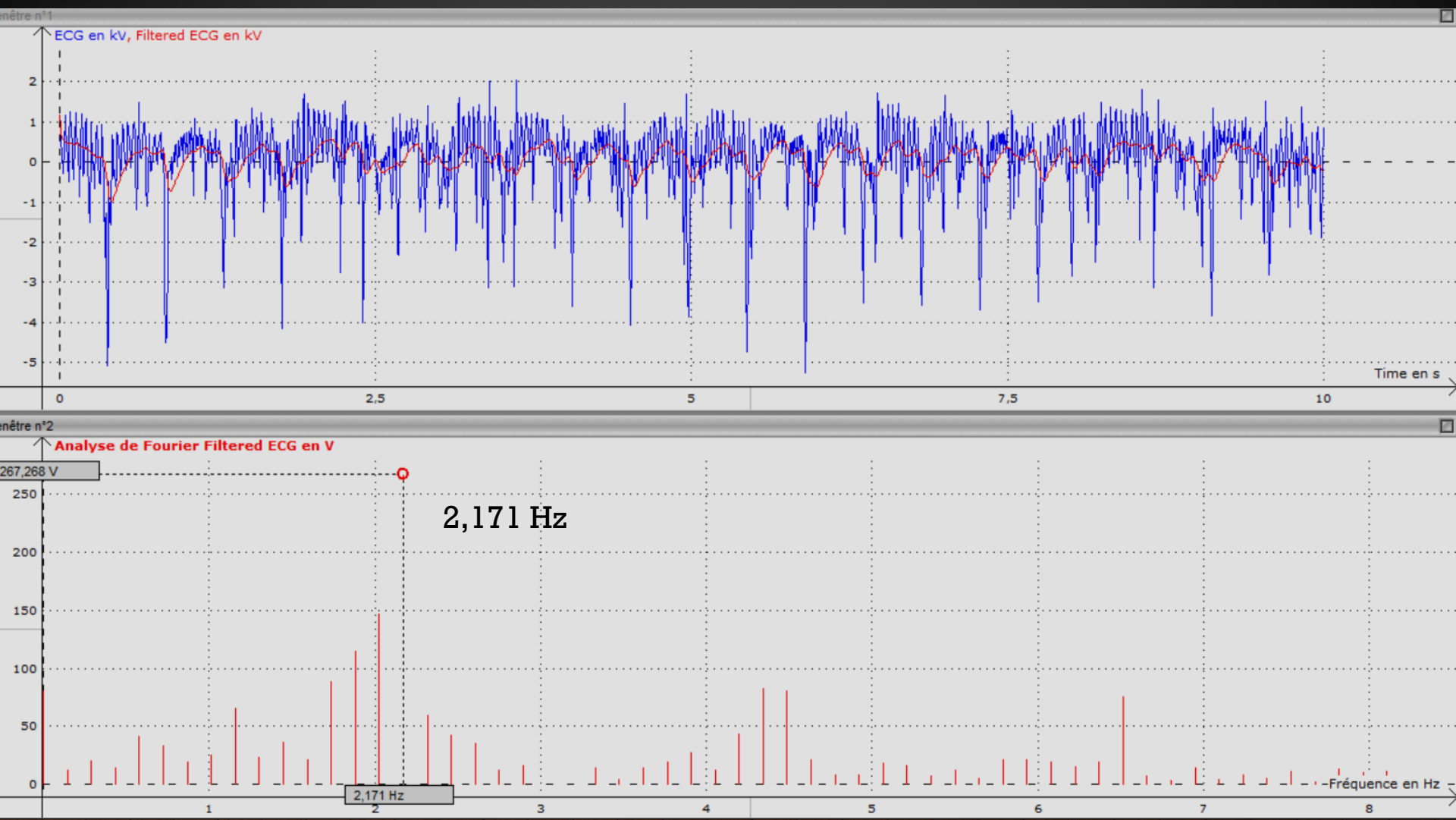


I 3 b: Signaux par-dessus plusieurs couches vestimentaires

Par-dessus un T-Shirt et un sweat: 2,074

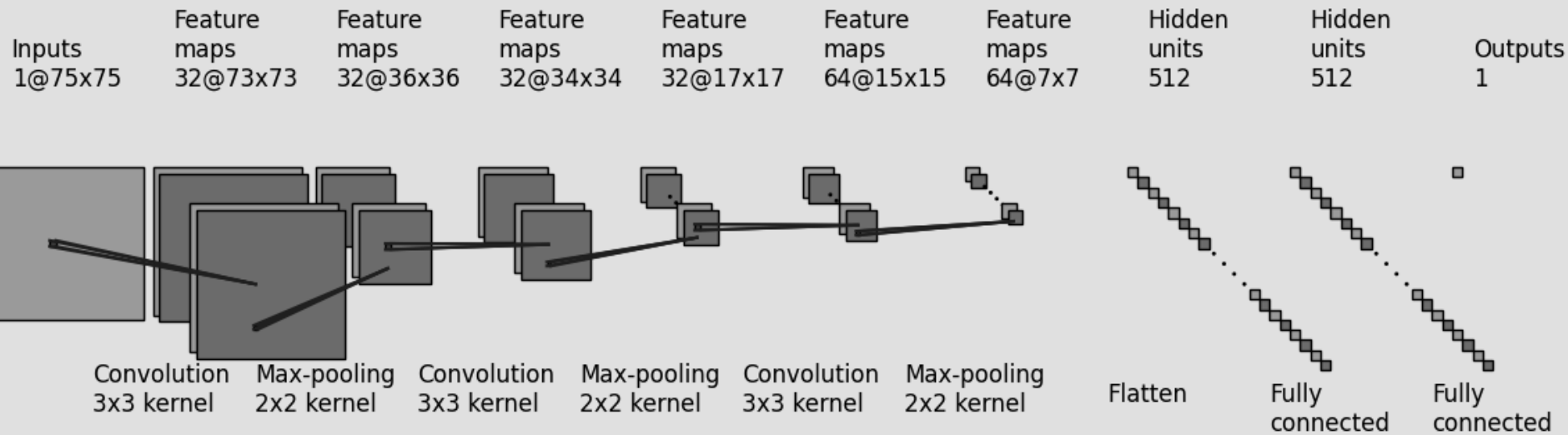


Par-dessus deux T-Shirt et un sweat: 2,171 Hz



I 3 c: Précision de la valeur mesurée de la fréquence cardiaque en fonction des couches vestimentaires

Quelle couche ?	Fréquence obtenue	Écart relatif par rapport à la valeur de référence
Directement sur la peau	1,64 Hz = 98 bpm	0 %
Par-dessus un T-Shirt	1,82 Hz = 109 bpm	11 %
Par-dessus deux T-Shirt	1,47 Hz = 88 bpm	10 %
Par-dessus un T-Shirt et un Sweat	2,07 Hz = 124 bpm	27 %
Par-dessus deux T-Shirt et un Sweat	2,17 Hz = 130 bpm	33 %



MRL Eye Dataset



II 2: Les paramètres choisis (Fonction d'erreur)

Erreur quadratique en moyenne
(Mean Squared Error) :

Formule pour 1 image

y_i : valeur théorique

\hat{y}_i : valeur prédite

$$(y_i - \hat{y}_i)^2$$

Entropie croisée en binaire
(Binary Crossentropy) :

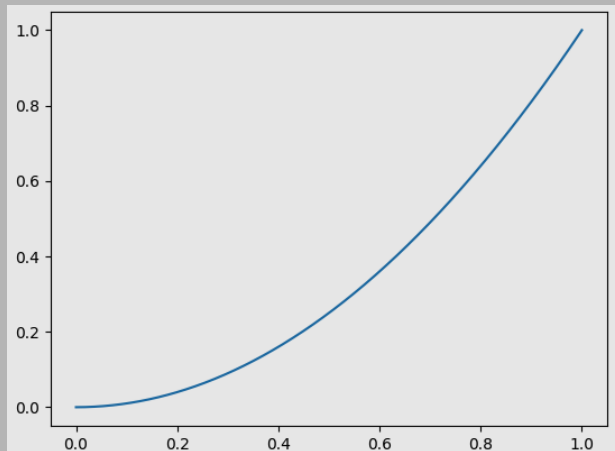
Formule pour 1 image

t : valeur théorique

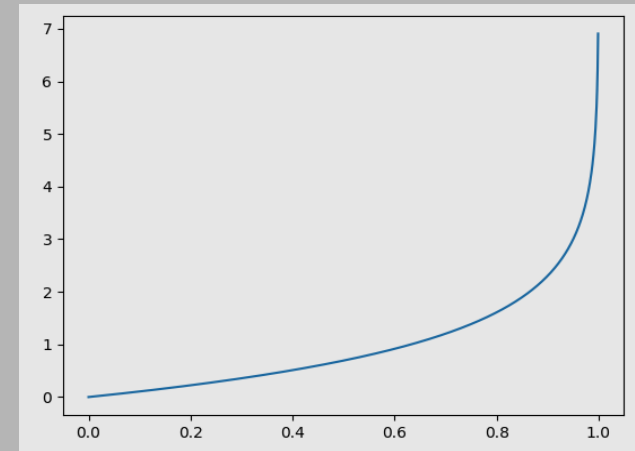
p : valeur prédite

$$-(t * \log(p) + (1 - t) * \log(1 - p))$$

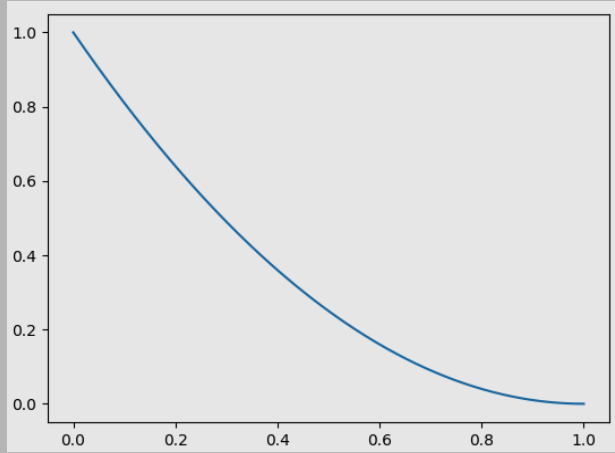
Pour la
valeur
théorique
0 :



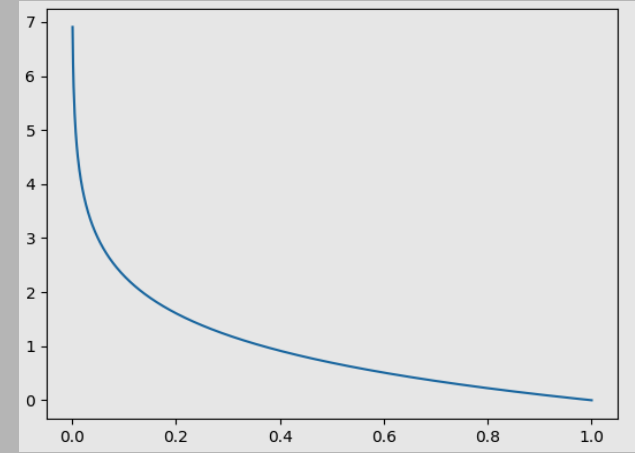
Pour la
valeur
théorique
0 :



Pour la
valeur
théorique
1 :



Pour la
valeur
théorique
1 :



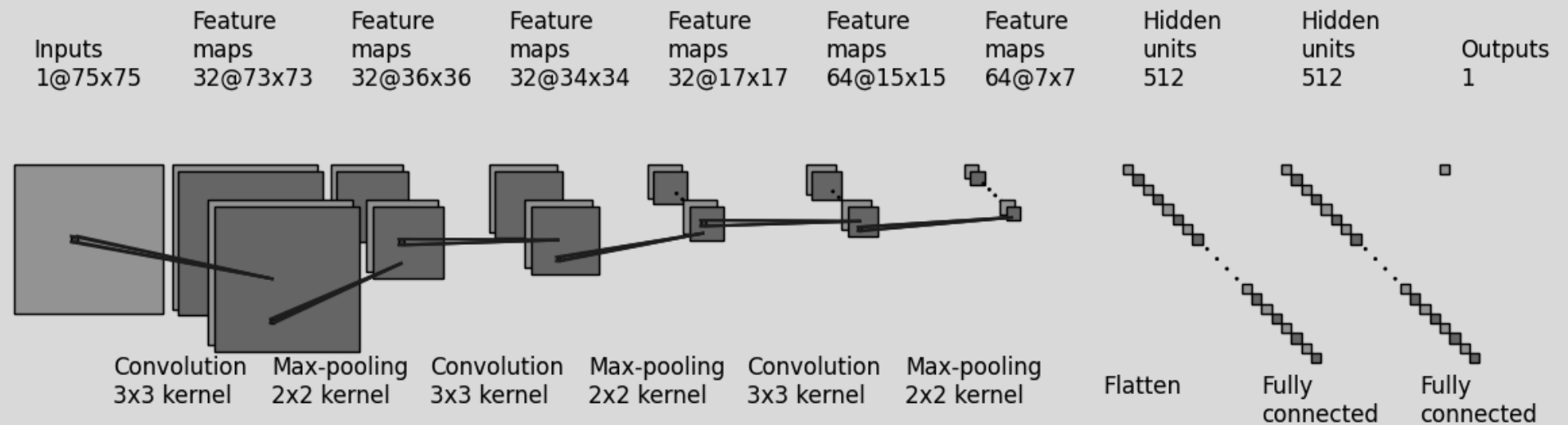
II 3: Les résultats de l'entraînement (1^{er} test sur 1000 images)

['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 9, 'test', 0.966]
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 2, 'epoch', 10, 'test', 0.961]
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 8, 'test', 0.96]
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 4, 'epoch', 8, 'test', 0.949]
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 6, 'test', 0.948]
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 2, 'epoch', 2, 'test', 0.942]
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 8, 'test', 0.939]
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 2, 'epoch', 4, 'test', 0.936]
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 4, 'epoch', 6, 'test', 0.936]
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 4, 'epoch', 10, 'test', 0.934]
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 4, 'epoch', 5, 'test', 0.932]
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 3, 'test', 0.931]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 8, 'epoch', 5, 'test', 0.927]
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 4, 'epoch', 2, 'test', 0.923]
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 5, 'test', 0.922]
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 2, 'epoch', 7, 'test', 0.922]
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 9, 'test', 0.922]
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 2, 'epoch', 2, 'test', 0.921]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 2, 'epoch', 6, 'test', 0.921]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 2, 'epoch', 10, 'test', 0.921]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 4, 'epoch', 5, 'test', 0.917]
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 10, 'test', 0.916]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 4, 'epoch', 2, 'test', 0.916]
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 4, 'epoch', 4, 'test', 0.916]

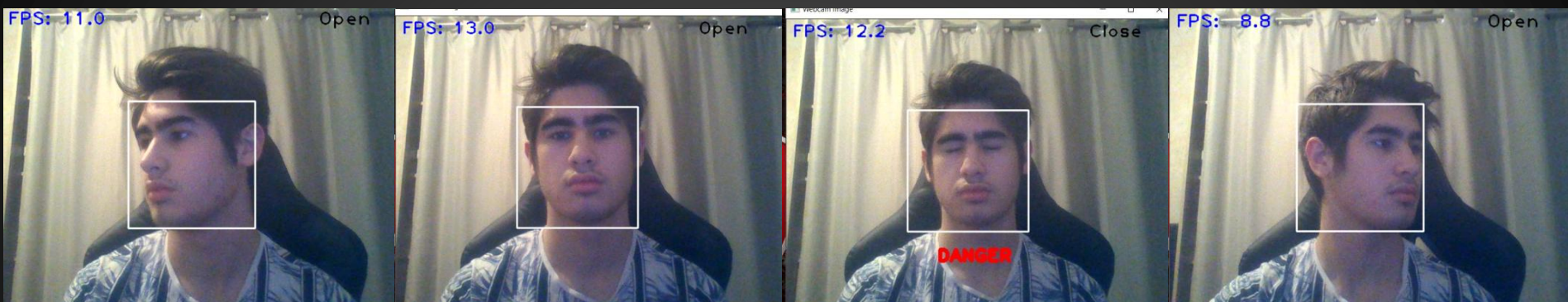
Parmi les 600 structures que j'ai entraînés, j'ai
décidé de ne considérer que les 13 meilleures.

II 3: Les résultats de l'entraînement

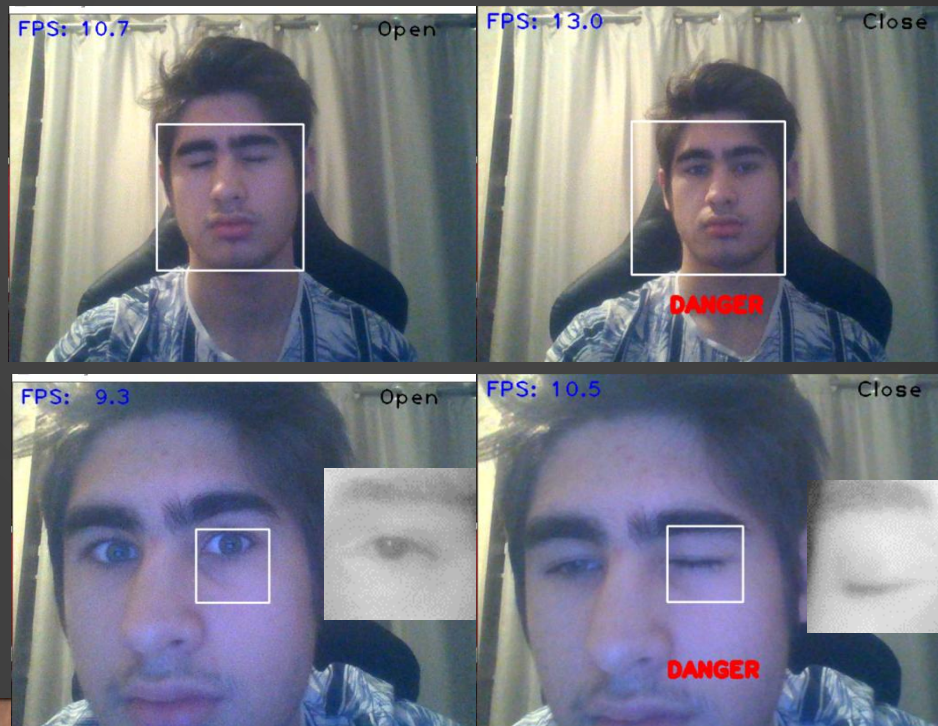
Structure	1 ^{er} test sur les 1000 images	2 ^{ème} test sur les 1000 images	Test sur les 24000 images	Moyenne des 3 tests
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 9]	0,966	0,948	0.924	<u>0,946</u>



Fonctionnement classique:



Mais des erreurs
trop fréquentes:



Des erreurs dues
à la base de
données choisie:

III : Les limites de chaque méthode

Limites de la méthode cardiaque:



$$\Delta t * \Delta f \simeq 1$$

Limites de la méthode informatique:



Annex es

1° Etude du filtre passe-bas

2° Création de la base de données au format Keras

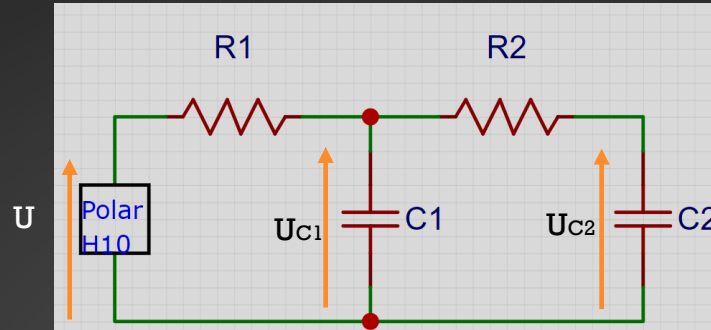
3° Entraînement des 600 réseaux de neurones et test sur 1000 images

4° Résultats des entraînements sur les 13 réseaux de neurones

4° Prédiction en temps réel

5° Images utilisées

Etude théorique du filtre passe-bas d'ordre 2



Etude temporelle

Lois des mailles dans les mailles 1 et 2:

$$RC * \dot{U}_{C2} + U_{C2} = U_{C1} \quad (1)$$

$$RC * (\dot{U}_{C2} + \dot{U}_{C1}) + U_{C1} = U \quad (2)$$

On dérive l'équation (1) et on utilise les deux équations pour simplifier:

$$\dot{U}_{C2} + RC * \ddot{U}_{C2} = \dot{U}_{C1}$$

$$\dot{U}_{C2} + RC * \ddot{U}_{C2} = \frac{1}{RC} * (U - U_{C1}) - \dot{U}_{C2} \quad \text{d'après (2)}$$

$$\dot{U}_{C2} + RC * \ddot{U}_{C2} = \frac{1}{RC} * (U - RC * \dot{U}_{C2} - U_{C2}) - \dot{U}_{C2} \quad \text{d'après (1)}$$

On obtient donc:

$$3 * RC * \dot{U}_{C2} + RC^2 * \ddot{U}_{C2} + U_{C2} = U$$

D'où:

$$\ddot{U}_{C2} + \frac{3}{RC} * \dot{U}_{C2} + \frac{1}{RC^2} * U_{C2} = \frac{1}{RC^2} * U$$

Il faut néanmoins passer de dérivée continue à dérivée discrète, on a ainsi:

$$\ddot{U}_{C2}[N] = \frac{d\dot{U}_{C2}}{dt}[N] = \frac{d\frac{U_{C2}[N] - U_{C2}[N-1]}{Te}}{dt}[N]$$

$$\ddot{U}_{C2}[N] = \frac{U_{C2}[N] - 2 * U_{C2}[N-1] + U_{C2}[N-2]}{Te^2}$$

On en déduit finalement:

$$U_{C2}[N] \frac{RC^2 + 3RC * Te + Te^2}{RC^2 * Te^2} = \frac{U[N]}{RC^2} + U_{C2}[N-1] \frac{2RC + 3Te}{RC * Te^2} - \frac{U_{C2}[N-2]}{Te^2}$$

Etude fréquentielle

Dans la partie temporelle, nous avons vu l'apparition du terme RC que l'on doit donc déterminer:

$$U_{C2} = \frac{\frac{1}{jC\omega}}{\frac{1}{jC\omega} + R} * U = \frac{1}{1 + jRC\omega} * U$$

L'expression de la fonction de transfert est alors:

$$H(j\omega) = \frac{1}{1 + jRC\omega}$$

On vérifie bien le caractère passe-bas d'ordre 2 du filtre:

$$|H(j\omega)| \underset{\omega \rightarrow 0}{\sim} 1 \text{ donc } G_{dB} \underset{\omega \rightarrow 0}{\rightarrow} 0$$

$$|H(j\omega)| \underset{\omega \rightarrow +\infty}{\sim} \frac{1}{RC\omega} \text{ donc } G_{dB} \underset{\omega \rightarrow +\infty}{\sim} -40\log(\omega) - 40\log(RC)$$

Il ne reste plus qu'à déterminer RC:

$$|H(j\omega_{coupure})| = \frac{1}{\sqrt{2}}$$

$$|1 + jRC\omega_{coupure}|^2 = \sqrt{2}$$

$$1 + RC^2\omega_{coupure}^2 = \sqrt{2}$$

$$RC\omega_{coupure} = \sqrt{\sqrt{2} - 1}$$

$$RC = \frac{\sqrt{\sqrt{2} - 1}}{2 * \pi * f_{coupure}}$$

Avec $f_{coupure} = 4Hz$ on trouve $RC = 2,56 * 10^{-2}s$

Résultats des entraînements sur les 13 réseaux de neurones

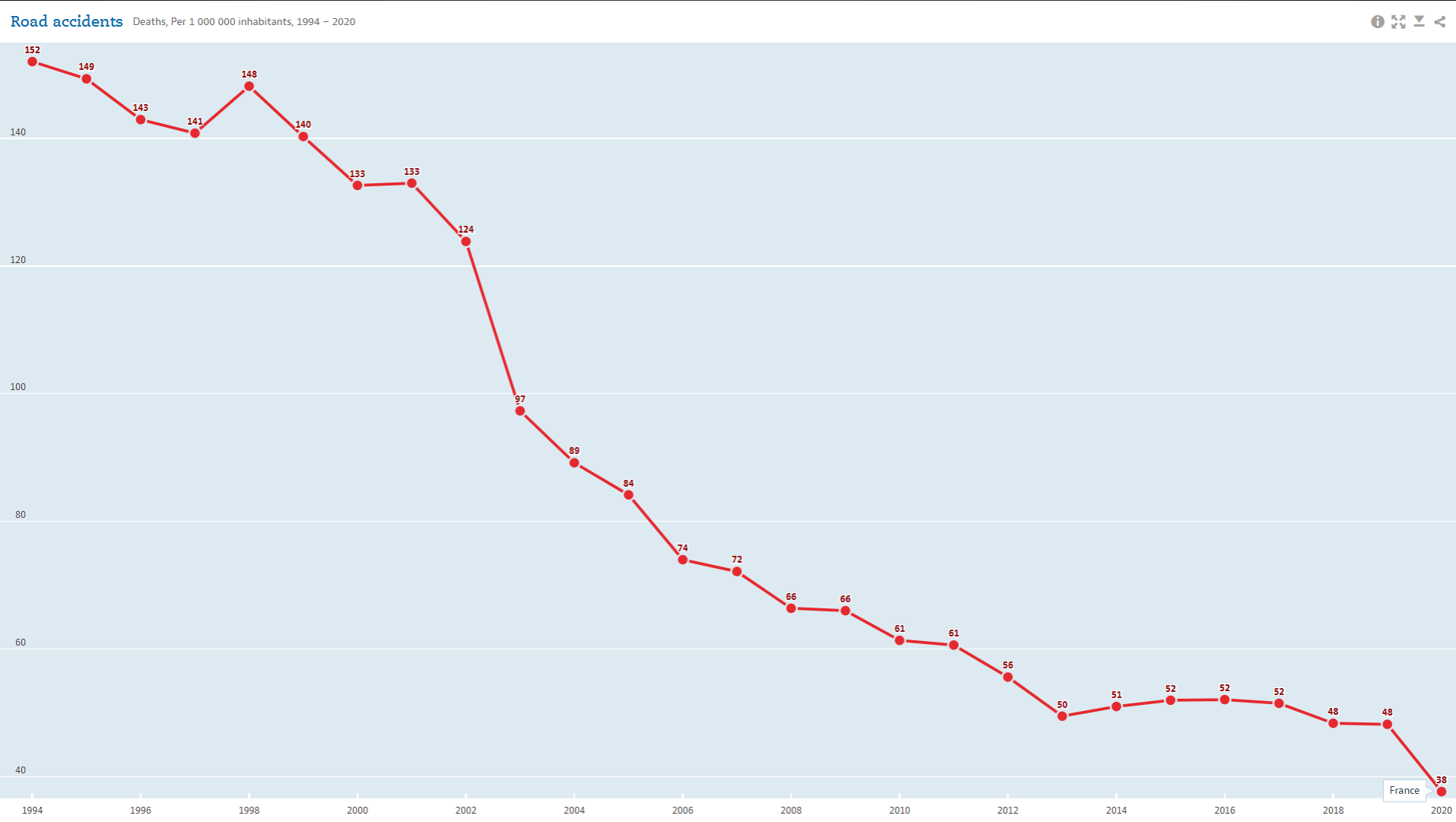
Structure	1 ^{er} test sur les 1000 images	2 ^{ème} test sur les 1000 images	Test sur les 24000 images	Moyenne des 3 tests
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 9]	0,966	0,948	0.92445833	0,94615278
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 2, 'epoch', 10]	0,961	0,812	0.84866667	0,87388889
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 8]	0,96	0,882	0.87820833	0,90673611
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 4, 'epoch', 8]	0,949	0,916	0.88320833	0,91606944
['convolution', 32, 32, 64, 'network', 512, 512, 'batch_size', 2, 'epoch', 6]	0,948	0,887	0.86145833	0,89881944
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 2, 'epoch', 2]	0,942	0,91	0.91541667	0,92247222
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 8]	0,939	0,809	0.84754167	0,86518056
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 2, 'epoch', 4]	0,936	0,834	0.882	0,884
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 4, 'epoch', 6]	0,936	0,914	0.91945833	0,92315278
['convolution', 32, 32, 64, 'network', 256, 256, 'batch_size', 4, 'epoch', 10]	0,934	0,528	0.696	0,71933333
['convolution', 32, 32, 64, 'network', 512, 256, 128, 64, 'batch_size', 4, 'epoch', 5]	0,932	0,904	0.91045833	0,91548611
['convolution', 32, 32, 64, 'network', 512, 256, 'batch_size', 4, 'epoch', 3]	0,931	0,927	0.917625	0,92520833
['convolution', 32, 32, 64, 'network', 256, 256, 128, 128, 'batch_size', 8, 'epoch', 5]	0,927	0,871	0.908875	0,90229167

Images utilisées



<https://www.capatv.com/portfolio-item/quand-le-sommeil-tue/>

Images utilisées



<https://data.oecd.org/transport/road-accidents.htm>

Images utilisées



37% DES 153 DÉCÈS
SURVENUS SUR AUTOROUTE EN 2011
ÉTAIENT DUS À LA SOMNOLENCE

137.2 01:03 79

Source : <https://www.maaf.fr/fr/assurance-auto/somnolence-au-volant-somme>

Images utilisées



<https://www.brusselstimes.com/news/eu-affairs/145084/motoring-organisations-advise-no-heavy-coats-inside-the-car/>

Images utilisées



Images utilisées



<https://www.auto-moto.com/pratique/reflexes-a-adopter-lorsquon-conduit-nuit-fp-162608.html>

II 2: Les paramètres choisis

(Couche de convolution, Fully-connected, Batch_size, Epochs)

Couches de convolution	1 ^{re} couche: 32 filtres de convolution	1 ^{re} couche: 32 filtres de convolution 2 ^{ème} couche: 64 filtres de convolution	1 ^{re} couche: 32 filtres 2 ^{ème} couche: 32 filtres 3 ^{ème} couche: 64 filtres
------------------------	---	---	--

Structure de la couche fully-connected	2 couches cachées: 1 ^{re} : 512 neurones 2 ^{ème} : 512 neurones	2 couches cachées: 1 ^{re} : 512 neurones 2 ^{ème} : 256 neurones	2 couches cachées: 1 ^{re} : 256 neurones 2 ^{ème} : 256 neurones	4 couches cachées: 1 ^{re} : 512 neurones 2 ^{ème} : 256 neurones 3 ^{ème} : 128 neurones 4 ^{ème} : 64 neurones	4 couches cachées: 1 ^{re} : 256 neurones 2 ^{ème} : 256 neurones 3 ^{ème} : 128 neurones 4 ^{ème} : 128 neurones
Batch_size	2	4	8	16	

Epochs	1	2	3	4	5	6	7	8	9	10
--------	---	---	---	---	---	---	---	---	---	----