



UNIVERSITY OF SCIENCE AND TECHNOLOGY HOUARI BOUMEDIENE

DEPARTMENT OF IT

IMPLEMENTAION OF A SCIENTIFC ARTICLE

GEOMETRIC FEATURES-BASED GASTROINTESTINAL TRACT DISEASES DETECTION

SUPERVISOR

I.SETITRA

MASTER CANDIDATE

CHEMLOUL MOUNIR

DJILLALI MAZIGH

FERDJI ELIAS

ACADEMIC YEAR

2021-2022

Contents

1	INTRODUCTION	1
1	PHASE DE PRÉ-TRAITEMENT	3
1.1	Allongement du contraste	3
1.2	Morphologie mathématique sur les images en niveau de gris	4
1.2.1	Dilatation	4
1.2.2	Érosion	5
1.3	Filtrage des parties claires de l'image	5
1.3.1	Ouverture	5
1.3.2	Chapeau haut de forme (Top Hat)	5
1.4	Filtrage des parties sombres de l'image	5
1.4.1	Fermeture	5
1.4.2	Chapeau bas de forme (Bottom Hat)	6
1.5	Finalisation du pré traitement	6
1.5.1	Amélioration visuelle	6
1.5.2	Correction des bruits	6
2	CIELAB ET SEGMENTATION	7
2.1	Espace colorimétrique Cielab	7
2.1.1	Définition Cielab	7
2.1.2	Transformation du système RGB au CIELAB	7
2.2	Segmentation	9
2.2.1	Calcul des caractéristiques	9
2.2.2	Extraction des zones importante	11
2.3	Récupération des résultats	12
3	CONCLUSION	13

1

Introduction

Les infections du tractus gastro-intestinal (GIT) telles que les ulcères, les saignements, les polypes, la maladie de Crohn et le cancer sont assez courantes aujourd'hui dans le monde entier. L'endoscopie par capsule sans fil (WCE) est un moyen efficace d'investigation des maladies gastro-intestinales.

Cependant, plusieurs défis existent encore dans ce domaine, tels que la forme, la couleur, la texture, la taille et l'irrégularité des lésions. Pour faire face à ces problèmes, plusieurs méthodes informatiques sont introduites dans l'informatique.

domaine de la vision, mais ils n'ont utilisé que des fonctionnalités artisanales qui ont produit plusieurs fois des prédictions erronées. Dans cette recherche, une nouvelle technique est appliquée basée sur la fusion de caractéristiques convolutionnelles profondes (CNN) et géométriques. Initialement, les régions de la maladie sont extraites d'images WCE données à l'aide d'une nouvelle approche appelée caractéristiques de couleur à contraste amélioré. Les caractéristiques géométriques sont extraites de la région de la maladie segmentée.

Par la suite, une fusion unique des caractéristiques CNN profondes VGG16 et VGG19 est effectuée sur la base du vecteur Euclidean Fisher. Les caractéristiques uniques sont fusionnées avec des caractéristiques géométriques qui sont ensuite alimentées par une approche d'entropie conditionnelle pour une meilleure sélection des caractéristiques.

— Muhammad Sharif, Muhammad Attique Khan, Muhammad Rashid, Mussarat Yasmin, Farhat Afza Urcun John Tanik (2019): Deep CNN and geometric features-based gastrointestinal tract diseases detection and classification from wireless capsule endoscopy images, Journal of Experimental Theoretical Artificial Intelligence, DOI: 10.1080/0952813X.2019.1572657

1

Phase de pré-traitement

1.1 ALLONGEMENT DU CONTRASTE

Le contraste stretching correspond à modifier les niveaux de gris d'une image suivant une courbe que l'on choisit. D'après (Negi Bhandari, 2014), cette transformation est une succession de 3 droites de pentes l, m et n . L'axe des x représente les niveaux de gris de notre image avant transformation et l'axe des y va représenter les niveaux de gris après transformation.

L'axe des x représente les niveaux de gris de notre image avant transformation et l'axe des y va représenter les nouveaux niveaux de gris:

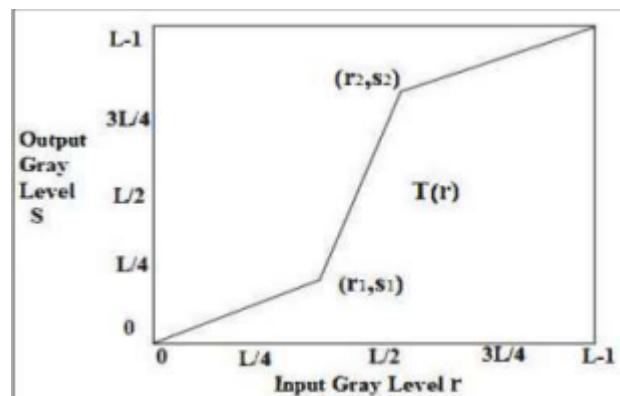


Figure 1.1: Graph of contrast stretching.

$$S = \begin{cases} lr & 0 \leq x < a \\ m(r - a) + la & a \leq x < b \\ n(rL + 1) + L1 & b \leq x < L-1 \end{cases} \quad (1.1)$$

- Si la pente (l, m, n) est égal à 1, aucune modification des niveaux de gris par rapport aux niveaux de gris initiaux

- Si la pente est supérieure à 1, les nouveaux niveaux de gris seront supérieurs à ceux qu'ils étaient avant (niveau de gris en entrée).
- Si la pente est inférieure à 1, les nouveaux niveaux de gris seront inférieurs à ceux qu'ils étaient avant (niveau de gris en entrée).

Ainsi des pentes < 1 assombriront l'image tandis que des valeurs supérieures à 1 l'éclaireront. Dans notre cas, on accentue le contraste entre zones sombres et claires. En pratique, voici l'algorithme :

Algorithm 1.1 Contrast Stretching

```

for  $i = 1$  vers  $nblignes$ 
  for  $i = 1$  vers  $nbcolumn$ 
    if  $0 \leq A[i,j] < a$ 
       $R[i,j] = l * A[i,j]$ 
    else
      if  $(a \leq A[i,j] < b)$ 
         $R[i,j] = m(A[i,j] - a) + la$ 
      else
         $R[i,j] = n(A[i,j] - L + 1) + L - 1$ 
      end if
    end if
  end for
end for

```

Ici Les pentes l et m sont choisie et la valeur n est calculée comme suit:

$$n = \frac{L - 1 - (m(b - a) + la)}{L - 1 - b} \quad (1.2)$$

A est la matrice a traiter et R la matrice résultat.

1.2 MORPHOLOGIE MATHÉMATIQUE SUR LES IMAGES EN NIVEAU DE GRIS

La morphologie mathématique pour des images en niveaux de gris avec un élément structurant plat revient aux opérations suivantes

1.2.1 DILATATION

Sur une image I , il suffit de déplacer un élément structurant de la forme d'une matrice 3×3 , chercher le max de cette fenêtre, ensuite il faut le remplacer dans le pixel du centre (dans les images binaires, il suffit d'avoir un 1 dans le masque pour mettre le pixel central a 1). Les niveaux de gris élevée ont tendance à s'étaler d'une distance de 1 autour de ce point.

$$D(i,j) = \max \{I(i+l, j+k), -1 \leq k, l \leq 1\} \quad (1.3)$$

1.2.2 ÉROSION

Sur une image I , il suffit de déplacer un masque 3×3 et chercher le min de cette fenêtre et le remplacer dans le centre. C'est l'inverse de la dilatation, cette opération revient à une dilatation des valeurs faibles.

$$E(i, j) = \min \{I(i + l, j + k), -1 \leq k, l \leq 1\} \quad (1.4)$$

1.3 FILTRAGE DES PARTIES CLAIRES DE L'IMAGE

1.3.1 OUVERTURE

L'ouverture est une opération qui permet d'adoucir une image en éliminant les structures claires (sur la fonction créée par l'intensité des pixels) qui sont plus étroites que l'élément structurant. Pour faire une ouverture sur une image, il faut appliquer une érosion suivie d'une dilatation.

$$O(i, j) = D(E(i, j)) \quad (1.5)$$

1.3.2 CHAPEAU HAUT DE FORME (TOP HAT)

Le Top Hat est une opération de morphologie mathématique qui sert à extraire les composantes contrastées d'une image. Plus précisément, grâce au Top Hat, nous pouvons récupérer les composantes claires en isolant les objets plus grands que l'élément structurant. Le Top Hat est la différence entre l'image initiale et l'ouverture de l'image.

$$TOH(i, j) = I(i, j) - O(i, j) \quad (1.6)$$

Le Top Hat additionné à l'image originale augmente le contraste des régions claires au premier plan.

$$IM(i, j) = I(i, j) + TOH(i, j) \quad (1.7)$$

1.4 FILTRAGE DES PARTIES SOMBRES DE L'IMAGE

1.4.1 FERMETURE

La fermeture est aussi une opération qui adoucit une image, mais à l'inverse de l'ouverture, cette opération élimine les structures sombres plus étroites que le rayon de l'élément structurant (1 car on a une matrice 3×3) élément structurant. Pour faire une fermeture sur une image, il faut appliquer une dilatation suivie d'une érosion.

$$F(i, j) = E(D(i, j)) \quad (1.8)$$

1.4.2 CHAPEAU BAS DE FORME (BOTTOM HAT)

A l'inverse du Top Hat, le Bottom Hat quant à lui sert à extraire les composantes sombres de l'image en isolant les objets plus petits que l'élément structurant. Le Bottom Hat est la différence entre la fermeture de l'image et l'image initiale.

$$BOH(i,j) = F(i,j)I(i,j) \quad (1.9)$$

le filtre bottom-hat additionné à l'image originale améliore le contraste des parties sombres qui aura pour effet de les assombrir, et ressortir les structures sombres par rapport au background.

$$IM(i,j) = I(i,j) + BOH(i,j) \quad (1.10)$$

1.5 FINALISATION DU PRÉ TRAITEMENT

1.5.1 AMÉLIORATION VISUELLE

$$Image(i,j) = I(i,j) + TOH + BOH(i,j) \quad (1.11)$$

Cette combinaison de ces deux images filtrées nous une image qui est quasiment prête pour le reste du traitement. Cependant nous notons souvent l'apparition de bruit qui empêcherait un traitement optimal.

1.5.2 CORRECTION DES BRUITS

Pour diminuer le bruit, nous utilisons un filtre médian classique dans les opérations de pré traitement, ce filtre consiste à faire passer un masque 3x3 sur l'image et remplacer le pixel du centre par la valeur médiane présente dans le masque.

$$IM(i,j) = med \{I(i+l, j+k), -1 \leq k, l \leq 1\} \quad (1.12)$$

2

Cielab et segmentation

2.1 ESPACE COLORIMÉTRIQUE CIELAB

La détection des maladies gastro-intestinales en se basant sur des images WCE (Endoscopie par capsule sans fil) est assez compliquée à cause des irrégularités sur les zones infectées, la similitude des couleurs et des tailles des zones malades.

Pour éviter tous ces problèmes on passe sur l'espace Cielab.

2.1.1 DÉFINITION CIELAB

Le CIELAB est un espace colorimétrique défini en 1976 par la CIE, chaque couleur perceptible dans cet espace est représentée par un point de couleur possédant les trois coordonnées $L^*a^*b^*$, soit L^* la clarté et a^*b^* l'écart de la couleur par rapport à celle d'une surface grise de même clarté.

Cet espace vise deux objectifs :

- La simplicité : l'utilisateur dispose de moyens plus simples pour interpréter les données
- L'uniformité : Les distances calculées entre les échantillons doivent correspondre de manière uniforme avec la différence visuelle entre les échantillons.

À ce niveau, notre image est dans l'espace colorimétrique RGB, et afin de passer à l'espace colorimétrique CIELAB nous devons passer par 5 étapes.

2.1.2 TRANSFORMATION DU SYSTÈME RGB AU CIELAB

ÉTAPE 1: NORMALISATION DES DONNÉES RGB (PASSAGE AU sRGB)

Les valeurs des composantes dans l'espace sRGB sont comprises entre 0 et 1. Il suffit donc de diviser les valeurs des plans Rouge, Vert et Bleu par la valeur maximum (255 pour 8bits). On se retrouve dans l'espace sRGB.

ÉTAPE 2: LINÉARISATION DES COULEURS

On notera les composantes linéaires par r,g,b. Ils sont obtenus par l'algorithme suivant :

Algorithm 2.1 Linéarisation de couleurs

```
if  $x \leq 0.4045$ 
   $x \leftarrow \frac{x}{12.92}$ 
else
   $x \leftarrow \left( \frac{x+0.055}{1.055} \right)^{2.4}$ 
end if
```

Pour linéariser les couleurs, il faudra exécuter cet algorithme sur les trois canaux r,g,b.

ÉTAPE 3: RGB LINÉAIRES VERS XYZ

Pour passer du rgb linéaire au XYZ, il faut réaliser une simple opération matricielle:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0,4124 & 0,3576 & 0,1805 \\ 0,2126 & 0,7152 & 0,0722 \\ 0,0193 & 0,1192 & 0,09505 \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix} \quad (2.1)$$

ce qui nous donne au final :

$$X = 0.4124r + 0.3576g + 0.1805b \quad (2.2)$$

$$Y = 0.2126r + 0.7152g + 0.0722b \quad (2.3)$$

$$Z = 0.0193r + 0.1192g + 0.9505b \quad (2.4)$$

ÉTAPE 4: CALIBRAGE DU SYSTÈME AVEC LE POINT BLANC.

Le point blanc, obtenu avec les trois composantes égales, correspond à l'illuminant D65. Donc en considérant $r = 1$, $g = 1$ et $b = 1$ et la matrice précédente (redonnée ci-après) on aura les composantes du point blanc noté X' , Y' et Z' :

$$X' = 0.4124 + 0.3576 + 0.1805 = 0.95046 \quad (2.5)$$

$$Y' = 0.2126 + 0.7152 + 0.0722 = 1.0 \quad (2.6)$$

$$Z' = 0.0193 + 0.1192 + 0.9505 = 1.08883 \quad (2.7)$$

ÉTAPE 5: TRANSFORMATION LAB

En utilisant les données de la troisième et quatrième étape nous pouvons calculer les coordonnées L, a et b comme suit:

- Calcul de L:

Algorithm 2.2 L

```
if  $(Y/Y') > 0,008856$ 
   $L \leftarrow 116.(Y/Y')^{1/3} - 16$ 
else
   $L \leftarrow 903.3(Y/Y')$ 
end if
```

- Calcul de a:

$$a = 500 (f(X/X') - f(Y/Y')) \quad (2.8)$$

- Calcul de b:

$$b = 200 (f(Z/Z') - f(Z'/Z')) \quad (2.9)$$

- Remarque: la fonction $f(I)$ est:

Algorithm 2.3 Fonction $f(I)$

```
if  $I > 0,008856$ 
   $F \leftarrow I^{1/3}$ 
else
  if  $t \leq 0,008856$ 
     $F \leftarrow 7,787t + (16/116)$ 
  end if
end if
```

2.2 SEGMENTATION

Après être passé à l'espace colorimétrique CIELAB, nous allons donc maintenant extraire certaines caractéristiques propres à chaque pixel à savoir (Moyenne, Variance, l'écart type, l'asymétrie et l'aplatissement). Ce travail est à faire sur les deux plans a et b car il s'agit de la chrominance.

2.2.1 CALCUL DES CARACTÉRISTIQUES

Voici comment extraire les caractéristiques précédentes:

MEAN (MOYENNE)

$$Mean = \sum_{i=1}^N \frac{f_i}{N} \quad (2.10)$$

VARIANCE

$$Variance = \sum_{i=1}^N \frac{(f_i - \bar{f})^2}{N - 1} \quad (2.11)$$

SD (STANDARD DEVIATION)

$$SD = \sqrt{Variance} \quad (2.12)$$

KURTOSIS (APLATISSEMENT)

$$Kurtosis = \frac{\sum_{i=1}^N \frac{(f_i - \bar{f})^4}{N}}{SD^4} \quad (2.13)$$

SKEWNESS (ASYMÉTRIE)

$$skewness = \frac{n}{(n-1)(n-2)} \sum_{i=1}^N \left(\frac{(f_i - \bar{f})}{SD} \right)^3 \quad (2.14)$$

FUSION

Après avoir récupéré les caractéristiques de tout les pixels, on se retrouve avec deux vecteurs de caractéristiques qui ont la même taille. Maintenant il faut fusionner ces deux vecteurs de la manière suivante.

$$\xi_{fused} = \sum_{i=1}^M [f_i] \quad (2.15)$$

MEAN DEVIATION (DÉVIATION)

La valeur absolue des écarts (Absolute average deviation ou Mean Deviation en anglais), on fait ce calcul sur la fusion des deux vecteurs de caractéristiques de cette manière :

$$\xi_{MD} = \sum_{i=1}^N \frac{|f_i - \bar{f}|}{N} \quad (2.16)$$

Afin d'obtenir une valeur que nous allons utiliser comme un seuil dans la prochaine partie.

2.2.2 EXTRACTION DES ZONES IMPORTANTE

Dans cette partie nous allons parler des méthodes qui vont nous permettre de créer nos cluster afin d'isoler la zone malade de la zone saine.

THRESHOLDING

Comme dit précédemment, nous allons utiliser le MD comme poids afin de faire un seuillage qui nous permettra de diviser nos éléments en deux segments C1 et C2 de cette manière là:

Algorithm 2.4 Première étape de seuillage

```
if  $valeur \geq seuil$ 
     $C1[] \leftarrow valeur$ 
else
     $C2[] \leftarrow valeur$ 
end if
```

CALCUL DE LA PROBABILITÉ MINIMALE

Après avoir classé nos caractéristiques, on calcule une probabilité conditionnelle à partir des deux clusters afin de tirer une valeur minimale.

$$P_c = P(f_i|f_j) = \frac{P(f_i \cap f_j)}{P(f_j)} \quad (2.17)$$

$$P(f_i \cap f_j) = P(f_i \cap f_j)P(f_j) \quad (2.18)$$

Supposons que la probabilité minimale trouvée est :

$$\xi_{lp}(f)$$

On va faire un thresholding qui cette fois ci va différencier entre les régions saines et les lésions. Le pseudo code du thresholding se présente de cette manière:

Algorithm 2.5 Thresholding

```
if  $f_{ij} \geq \xi_{lp}(f)$ 
     $TH[i,j] \leftarrow 255$ 
else
     $TH[i,j] \leftarrow 100$ 
end if
```

Il nous donnera en sortie une image binarisée

DATASET

La dernière étape consiste à faire un dataset avec les images enregistrées afin de les classer plus tard dans la deuxième partie du projet. La première étape après avoir récupéré l'image binarisée lors de la dernière étape. En premier lieu nous allons inverser l'image:

$$INV(i,j) = |TH(i,j) - 255|$$

Ensuite il faut raffiner l'image en utilisant les deux opérations morphologiques de dilatation et de remplissage (fillings). Enfin maintenant il faut convertir cette image en RGB pour récupérer ses couleurs originales.

2.3 RÉCUPÉRATION DES RÉSULTATS

Enfin, on sort avec l'image segmentée et on récupère des informations sur la lésion en calculant sa surface, la longueur maximale et minimale de l'axe, son orientation, son excentricité, sa longueur et sa largeur. Ces caractéristiques seront stockées dans une matrice qu'on utilisera dans la classification.

3

Conclusion

Pour résumer ce que nous avons compris de cet article, nous comprenons qu'il s'agit d'une phase de pré-classification où il faudra à partir d'une image, localiser une partie malade.

Pour faire ça, nous commençons par une phase de pré-traitement qui nous permettra d'améliorer le contraste de l'image avant de passer vers un espace colorimétrique où les différences de couleurs entre les parties saines et malades seront plus visibles, nous les segmenterons et les classerons en fonction de leurs couleurs pour en faire un dataset qui servira à la classification.

Pour le pré-traitement, on applique l'algorithme de contrast stretching qui va nous permettre d'accentuer le contraste entre les zones sombres et claires. Ensuite les deux algorithmes top hat et bottom hat quant à eux vont nous permettre d'améliorer le contraste sur les pics clairs ainsi que les pics sombres de l'image. Pour finir on applique un filtre médian pour éliminer les bruits (les petites valeurs isolées dans la matrice). Et maintenant que l'image est prête nous pouvons passer au traitement.

Au niveau du traitement, nous passons au CIELAB pour avoir une compréhension des données plus simple. Afin d'extraire les zones importantes dans l'image il faut extraire les vecteurs de caractéristiques de chacun des plans de chrominance (a et b) avant de les fusionner. Après cette fusion, il faut calculer le MD qui va nous permettre de créer deux clusters différents.

A partir de ces deux clusters on calcule une probabilité minimale qui va nous permettre de faire un seuillage entre les parties saines et les parties malades.

Pour finir on fait un dernier traitement sur l'image pour pouvoir créer notre dataset.