collageradiomics

Release 0.2

BrIC Laboratory

Sep 07, 2022

CONTENTS:

CoLlAGe captures subtle anisotropic differences in disease pathologies by measuring entropy of co-occurrences of voxel-level gradient orientations on imaging computed within a local neighborhood.

CoLlAGe is based on the hypothesis that disruption in tissue microarchitecture can be quantified on imaging by measuring the disorder in voxel-wise gradient orientations. CoLlAGe involves assigning every image voxel a 'disorder value' associated with the co-occurrence matrix of gradient orientations computed around every voxel.

Details on extraction of CoLlAGe features are included in [1]. After feature extraction, the subsequent distribution or different statistics such as mean, median, variance etc can be computed and used in conjunction with a machine learning classifier to distinguish similar appearing pathologies. The feasibility of CoLlAGe in distinguishing cancer from treatment confounders/benign conditions and characterizing molecular subtypes of cancers has been demonstrated in the context of multiple challenging clinical problems.

CONTENTS: 1

HELPFUL LINKS

Instructions: README

RadxTools Website: https://radxtools.github.io

Original Paper: Co-occurrence of Local Anisotropic Gradient Orientations (CoLlAGe): A new radiomics descriptor

CODE DOCUMENTATION

2.1 Notes

The **attributes** below represent the public output intended to be available to consumers of this module.

class collageradiomics.**Collage**(*img_array*, *mask_array*, *svd_radius=5*, *verbose_logging=False*, *cooccurence_angles=[0.0, 0.7853981633974483, 1.5707963267948966, 2.356194490192345, 3.141592653589793, 3.9269908169872414, 4.71238898038469, 5.497787143782138]*, *difference_variance_interpretation=DifferenceVarianceInterpretation.XMinusYVariance*, *haralick_window_size=-1*, *num_unique_angles=64*)

    Bases: `object`

    This is the main object in the Collage calculation system. Usage: create a Collage object and then call the `execute()` function.

        Parameters

- **image_array** (`numpy.ndarray`) – image to run collage upon

- **mask_array** (`numpy.ndarray`) – mask that correlates with the image

- **svd_radius** (`int, optional`) – radius of svd. Defaults to 5.

- **verbose_logging** (`bool, optional`) – This parameter is now ignored. Please use the python logging module.

- **cooccurence_angles** (`list, optional`) – list of angles to use in the cooccurence matrix. Defaults to [x*numpy.pi/4 for x in range(8)]

- **difference_variance_interpretation** (`DifferenceVarianceInterpretation, optional`) – Feature 10 has two interpretations, as the variance of **|x-y|** or as the variance of P(**|x-y|**).].Defaults to DifferenceVarianceInterpretation.XMinusYVariance.

- **haralick_window_size** (`int, optional`) – size of rolling window for texture calculations. Defaults to -1.

- **num_unique_angles** (`int, optional`) – number of bins to use for the texture calculation. Defaults to 64.

        Attributes

    collage_output
        Array representing collage upon the mask within the full images.

    cooccurence_angles
        Iterable of angles that will be used in the cooccurence matrix.

difference_variance_interpretation
>   Feature 10 has two interpretations, as the variance of **|x-y|** or as the variance of P(**|x-y|**).].

haralick_window_size
>   Number of pixels around each pixel to calculate a haralick texture.

img_array
>   The original image.

is_3D
>   Whether we are using 3D collage calculations (True) or 2D (False)

mask_array
>   Array passed into Collage.

num_unique_angles
>   Number of bins to use for texture calculations.

svd_radius
>   SVD radius is used to calculate the pixel radius for the dominant angle calculation.

verbose_logging
>   This parameter is now ignored.

Methods

execute()	Begins haralick calculation.
get_single_feature_output(which_feature)	Output a single collage output feature.

property collage_output

>   Array representing collage upon the mask within the full images. If the input was 2D, the output will be height×width×13 where "13" is the number of haralick textures. If the input was 3D, the output will be height×width×depth×13x2 where "2" is the primary angle (element 0) or the secondary angle (element 1)

>   The output will have numpy.nan values everywhere outside the masked region.

>   > **Getter**
>   >>   Returns array the same shape as the original image with collage in the mask region.

>   > **Type**
>   >>   numpy.ndarray

property cooccurence_angles

>   Iterable of angles that will be used in the cooccurence matrix.

>   > **Getter**
>   >>   Returns the Iterable of cooccurence angles.

>   > **Setter**
>   >>   Sets the angles to be used in the cooccurence matrix.

>   > **Type**
>   >>   int

property difference_variance_interpretation

>   Feature 10 has two interpretations, as the variance of **|x-y|** or as the variance of P(**|x-y|**).]. Defaults to DifferenceVarianceInterpretation.XMinusYVariance.

>   > **Getter**
>   >>   Returns requested variance interpretation.

> **Setter**
>> Sets requested variance interpretation.
>
> **Type**
>> *DifferenceVarianceInterpretation*

execute()

> Begins haralick calculation.
>
>> **Returns**
>>> An image at original size that only has the masked section filled in with collage calculations.
>>
>> **Return type**
>>> numpy.ndarray

get_single_feature_output(*which_feature*)

> Output a single collage output feature. If this was a 3D calculation, the output will be of size height×width×depth×2 where the "2" represents the collage calculation from the primary angle (0) or secondary angle (1).
>
>> **param which_feature**
>>> Either an integer from 0 to 12 (inclusive) or a HaralickFeature enum value
>
> :type which_feature HaralickFeature

property haralick_window_size

> Number of pixels around each pixel to calculate a haralick texture.
>
>> **Getter**
>>> Returns requested number of pixels.
>>
>> **Setter**
>>> Sets requested number of pixels.
>>
>> **Type**
>>> int

property img_array

> The original image.
>
>> **Getter**
>>> Returns the original image array.
>>
>> **Setter**
>>> Sets the original image array.
>>
>> **Type**
>>> np.ndarray

property is_3D

> Whether we are using 3D collage calculations (True) or 2D (False)

property mask_array

> Array passed into Collage.
>
>> **Getter**
>>> Returns the original mask array.
>>
>> **Setter**
>>> Sets the original mask array.
>>
>> **Type**
>>> np.ndarray

property num_unique_angles

> Number of bins to use for texture calculations. Defaults to 64.
>
> > **Getter**
> >
> > > Returns requested number of unique angles to bin into.
> >
> > **Type**
> >
> > > int

property svd_radius

> SVD radius is used to calculate the pixel radius for the dominant angle calculation.
>
> > **Getter**
> >
> > > Returns the SVD radius.
> >
> > **Setter**
> >
> > > Sets the SVD radius.
> >
> > **Type**
> >
> > > int

property verbose_logging

> This parameter is now ignored. Please use the python logging module.
>
> > **Getter**
> >
> > > Returns True if on.
> >
> > **Setter**
> >
> > > Turns verbose logging off or on.
> >
> > **Type**
> >
> > > bool

class collageradiomics.**DifferenceVarianceInterpretation**(*value*)

> Bases: Enum
>
> Feature 10 has two interpretations, as the variance of **|x-y|** or as the variance of P(**|x-y|**). See: https://ieeexplore.ieee.org/document/4309314
>
> > **Parameters**
> >
> > > **Enum** (DifferenceVarianceInterpretation) – Enumeration Helper For Haralick Features
>
> **ProbabilityXMinusYVariance = 1**
>
> **XMinusYVariance = 0**

class collageradiomics.**HaralickFeature**(*value*)

> Bases: IntEnum
>
> Enumeration Helper For Haralick Features
>
> > **Parameters**
> >
> > > **IntEnum** (HaralickFeature) – Enumeration Helper For Haralick Features
>
> **AngularSecondMoment = 0**
>
> **Contrast = 1**
>
> **Correlation = 2**
>
> **DifferenceEntropy = 9**

```
DifferenceVariance = 8

Entropy = 7

InformationMeasureOfCorrelation1 = 10

InformationMeasureOfCorrelation2 = 11

MaximalCorrelationCoefficient = 12

SumAverage = 4

SumEntropy = 6

SumOfSquareVariance = 3

SumVariance = 5
```

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

C

INDEX