JavaScript is disabled on your browser.

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Tree](http://docs.google.com/package-tree.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* Prev Class
* [Next Class](http://docs.google.com/org/opencv/features2d/FastFeatureDetector.html)
* [Frames](http://docs.google.com/index.html?org/opencv/features2d/DescriptorMatcher.html)
* [No Frames](http://docs.google.com/DescriptorMatcher.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Field](#3znysh7) |
* Constr |
* [Method](#2et92p0)
* Detail:
* [Field](#1t3h5sf) |
* Constr |
* [Method](#35nkun2)

org.opencv.features2d

## Class DescriptorMatcher

* java.lang.Object
  + [org.opencv.core.Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html)
    - org.opencv.features2d.DescriptorMatcher
* Direct Known Subclasses: [BFMatcher](http://docs.google.com/org/opencv/features2d/BFMatcher.html), [FlannBasedMatcher](http://docs.google.com/org/opencv/features2d/FlannBasedMatcher.html)  
    
  public class DescriptorMatcher  
  extends [Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html)  
  Abstract base class for matching keypoint descriptors. It has two groups of match methods: for matching descriptors of an image with another image or with an image set.

### Field SummaryFields

| Modifier and Type | Field and Description |
| --- | --- |
| static int | [**BRUTEFORCE**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#BRUTEFORCE) |
| static int | [**BRUTEFORCE\_HAMMING**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#BRUTEFORCE_HAMMING) |
| static int | [**BRUTEFORCE\_HAMMINGLUT**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#BRUTEFORCE_HAMMINGLUT) |
| static int | [**BRUTEFORCE\_L1**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#BRUTEFORCE_L1) |
| static int | [**BRUTEFORCE\_SL2**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#BRUTEFORCE_SL2) |
| static int | [**FLANNBASED**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#FLANNBASED) |

### Method SummaryMethods

| Modifier and Type | Method and Description |
| --- | --- |
| static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) | [**\_\_fromPtr\_\_**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#__fromPtr__(long))(long addr) |
| void | [**add**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#add(java.util.List))(java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> descriptors) Adds descriptors to train a CPU(trainDescCollectionis) or GPU(utrainDescCollectionis) descriptor collection. |
| void | [**clear**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#clear())() Clears the train descriptor collections. |
| [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) | [**clone**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#clone())() Clones the matcher. |
| [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) | [**clone**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#clone(boolean))(boolean emptyTrainData) Clones the matcher. |
| static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) | [**create**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#create(int))(int matcherType) |
| static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) | [**create**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#create(java.lang.String))(java.lang.String descriptorMatcherType) Creates a descriptor matcher of a given type with the default parameters (using default constructor). |
| boolean | [**empty**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#empty())() Returns true if there are no train descriptors in the both collections. |
| java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> | [**getTrainDescriptors**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#getTrainDescriptors())() Returns a constant link to the train descriptor collection trainDescCollection . |
| boolean | [**isMaskSupported**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#isMaskSupported())() Returns true if the descriptor matcher supports masking permissible matches. |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20java.util.List,%20int))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k) |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20java.util.List,%20int,%20java.util.List))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks) |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20java.util.List,%20int,%20java.util.List,%20boolean))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks, boolean compactResult) |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20int))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k) Finds the k best matches for each descriptor from a query set. |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20int,%20org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) Finds the k best matches for each descriptor from a query set. |
| void | [**knnMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#knnMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20int,%20org.opencv.core.Mat,%20boolean))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask, boolean compactResult) Finds the k best matches for each descriptor from a query set. |
| void | [**match**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#match(org.opencv.core.Mat,%20org.opencv.core.Mat,%20org.opencv.core.MatOfDMatch))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches) Finds the best match for each descriptor from a query set. |
| void | [**match**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#match(org.opencv.core.Mat,%20org.opencv.core.Mat,%20org.opencv.core.MatOfDMatch,%20org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) Finds the best match for each descriptor from a query set. |
| void | [**match**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#match(org.opencv.core.Mat,%20org.opencv.core.MatOfDMatch))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches) |
| void | [**match**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#match(org.opencv.core.Mat,%20org.opencv.core.MatOfDMatch,%20java.util.List))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks) |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20java.util.List,%20float))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance) |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20java.util.List,%20float,%20java.util.List))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks) |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20java.util.List,%20float,%20java.util.List,%20boolean))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks, boolean compactResult) |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20float))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance) For each query descriptor, finds the training descriptors not farther than the specified distance. |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20float,%20org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) For each query descriptor, finds the training descriptors not farther than the specified distance. |
| void | [**radiusMatch**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#radiusMatch(org.opencv.core.Mat,%20org.opencv.core.Mat,%20java.util.List,%20float,%20org.opencv.core.Mat,%20boolean))([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask, boolean compactResult) For each query descriptor, finds the training descriptors not farther than the specified distance. |
| void | [**read**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#read(java.lang.String))(java.lang.String fileName) |
| void | [**train**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#train())() Trains a descriptor matcher Trains a descriptor matcher (for example, the flann index). |
| void | [**write**](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html#write(java.lang.String))(java.lang.String fileName) |

### Methods inherited from class org.opencv.core.[**Algorithm**](http://docs.google.com/org/opencv/core/Algorithm.html)[getDefaultName](http://docs.google.com/org/opencv/core/Algorithm.html#getDefaultName()), [getNativeObjAddr](http://docs.google.com/org/opencv/core/Algorithm.html#getNativeObjAddr()), [save](http://docs.google.com/org/opencv/core/Algorithm.html#save(java.lang.String))

### Methods inherited from class java.lang.Objectequals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Field Detail

#### BRUTEFORCE public static final int BRUTEFORCESee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.BRUTEFORCE)

#### BRUTEFORCE\_HAMMING public static final int BRUTEFORCE\_HAMMINGSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.BRUTEFORCE_HAMMING)

#### BRUTEFORCE\_HAMMINGLUT public static final int BRUTEFORCE\_HAMMINGLUTSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.BRUTEFORCE_HAMMINGLUT)

#### BRUTEFORCE\_L1 public static final int BRUTEFORCE\_L1See Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.BRUTEFORCE_L1)

#### BRUTEFORCE\_SL2 public static final int BRUTEFORCE\_SL2See Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.BRUTEFORCE_SL2)

#### FLANNBASED public static final int FLANNBASEDSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.features2d.DescriptorMatcher.FLANNBASED)

### Method Detail

#### \_\_fromPtr\_\_ public static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) \_\_fromPtr\_\_(long addr)

#### add public void add(java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> descriptors) Adds descriptors to train a CPU(trainDescCollectionis) or GPU(utrainDescCollectionis) descriptor collection. If the collection is not empty, the new descriptors are added to existing train descriptors.Parameters:descriptors - Descriptors to add. Each descriptors[i] is a set of descriptors from the same train image.

#### clear public void clear() Clears the train descriptor collections.**Overrides:** [clear](http://docs.google.com/org/opencv/core/Algorithm.html#clear()) in class [Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html)

#### clone public [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) clone() Clones the matcher. that is, copies both parameters and train data. If emptyTrainData is true, the method creates an object copy with the current parameters but with empty train data.**Overrides:** clone in class java.lang.Object Returns:automatically generated

#### clone public [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) clone(boolean emptyTrainData) Clones the matcher.Parameters:emptyTrainData - If emptyTrainData is false, the method creates a deep copy of the object, that is, copies both parameters and train data. If emptyTrainData is true, the method creates an object copy with the current parameters but with empty train data. Returns:automatically generated

#### create public static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) create(int matcherType)

#### create public static [DescriptorMatcher](http://docs.google.com/org/opencv/features2d/DescriptorMatcher.html) create(java.lang.String descriptorMatcherType) Creates a descriptor matcher of a given type with the default parameters (using default constructor).Parameters:descriptorMatcherType - Descriptor matcher type. Now the following matcher types are supported:

* + - * BruteForce (it uses L2 )
      * BruteForce-L1
      * BruteForce-Hamming
      * BruteForce-Hamming(2)
      * FlannBased

Returns:automatically generated

#### empty public boolean empty() Returns true if there are no train descriptors in the both collections.**Overrides:** [empty](http://docs.google.com/org/opencv/core/Algorithm.html#empty()) in class [Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html) Returns:automatically generated

#### getTrainDescriptors public java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> getTrainDescriptors() Returns a constant link to the train descriptor collection trainDescCollection .Returns:automatically generated

#### isMaskSupported public boolean isMaskSupported() Returns true if the descriptor matcher supports masking permissible matches.Returns:automatically generated

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k)Parameters:queryDescriptors - Query set of descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total. descriptors and stored train descriptors from the i-th image trainDescCollection[i]. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks)Parameters:queryDescriptors - Query set of descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total.masks - Set of masks. Each masks[i] specifies permissible matches between the input query descriptors and stored train descriptors from the i-th image trainDescCollection[i]. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks, boolean compactResult)Parameters:queryDescriptors - Query set of descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total.masks - Set of masks. Each masks[i] specifies permissible matches between the input query descriptors and stored train descriptors from the i-th image trainDescCollection[i].compactResult - Parameter used when the mask (or masks) is not empty. If compactResult is false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k) Finds the k best matches for each descriptor from a query set.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object. descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors. These extended variants of DescriptorMatcher::match methods find several best matches for each query descriptor. The matches are returned in the distance increasing order. See DescriptorMatcher::match for the details about query and train descriptors.

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) Finds the k best matches for each descriptor from a query set.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.mask - Mask specifying permissible matches between an input query and train matrices of descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors. These extended variants of DescriptorMatcher::match methods find several best matches for each query descriptor. The matches are returned in the distance increasing order. See DescriptorMatcher::match for the details about query and train descriptors.

#### knnMatch public void knnMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, int k, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask, boolean compactResult) Finds the k best matches for each descriptor from a query set.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.mask - Mask specifying permissible matches between an input query and train matrices of descriptors.matches - Matches. Each matches[i] is k or less matches for the same query descriptor.k - Count of best matches found per each query descriptor or less if a query descriptor has less than k possible matches in total.compactResult - Parameter used when the mask (or masks) is not empty. If compactResult is false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors. These extended variants of DescriptorMatcher::match methods find several best matches for each query descriptor. The matches are returned in the distance increasing order. See DescriptorMatcher::match for the details about query and train descriptors.

#### match public void match([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches) Finds the best match for each descriptor from a query set.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.matches - Matches. If a query descriptor is masked out in mask , no match is added for this descriptor. So, matches size may be smaller than the query descriptors count. descriptors. In the first variant of this method, the train descriptors are passed as an input argument. In the second variant of the method, train descriptors collection that was set by DescriptorMatcher::add is used. Optional mask (or masks) can be passed to specify which query and training descriptors can be matched. Namely, queryDescriptors[i] can be matched with trainDescriptors[j] only if mask.at<uchar>(i,j) is non-zero.

#### match public void match([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) Finds the best match for each descriptor from a query set.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.matches - Matches. If a query descriptor is masked out in mask , no match is added for this descriptor. So, matches size may be smaller than the query descriptors count.mask - Mask specifying permissible matches between an input query and train matrices of descriptors. In the first variant of this method, the train descriptors are passed as an input argument. In the second variant of the method, train descriptors collection that was set by DescriptorMatcher::add is used. Optional mask (or masks) can be passed to specify which query and training descriptors can be matched. Namely, queryDescriptors[i] can be matched with trainDescriptors[j] only if mask.at<uchar>(i,j) is non-zero.

#### match public void match([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches)Parameters:queryDescriptors - Query set of descriptors.matches - Matches. If a query descriptor is masked out in mask , no match is added for this descriptor. So, matches size may be smaller than the query descriptors count. descriptors and stored train descriptors from the i-th image trainDescCollection[i].

#### match public void match([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html) matches, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks)Parameters:queryDescriptors - Query set of descriptors.matches - Matches. If a query descriptor is masked out in mask , no match is added for this descriptor. So, matches size may be smaller than the query descriptors count.masks - Set of masks. Each masks[i] specifies permissible matches between the input query descriptors and stored train descriptors from the i-th image trainDescCollection[i].

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance)Parameters:queryDescriptors - Query set of descriptors.matches - Found matches.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)! descriptors and stored train descriptors from the i-th image trainDescCollection[i]. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks)Parameters:queryDescriptors - Query set of descriptors.matches - Found matches.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)!masks - Set of masks. Each masks[i] specifies permissible matches between the input query descriptors and stored train descriptors from the i-th image trainDescCollection[i]. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, java.util.List<[Mat](http://docs.google.com/org/opencv/core/Mat.html)> masks, boolean compactResult)Parameters:queryDescriptors - Query set of descriptors.matches - Found matches.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)!masks - Set of masks. Each masks[i] specifies permissible matches between the input query descriptors and stored train descriptors from the i-th image trainDescCollection[i].compactResult - Parameter used when the mask (or masks) is not empty. If compactResult is false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance) For each query descriptor, finds the training descriptors not farther than the specified distance.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.matches - Found matches. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)! descriptors. For each query descriptor, the methods find such training descriptors that the distance between the query descriptor and the training descriptor is equal or smaller than maxDistance. Found matches are returned in the distance increasing order.

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask) For each query descriptor, finds the training descriptors not farther than the specified distance.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.matches - Found matches. false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)!mask - Mask specifying permissible matches between an input query and train matrices of descriptors. For each query descriptor, the methods find such training descriptors that the distance between the query descriptor and the training descriptor is equal or smaller than maxDistance. Found matches are returned in the distance increasing order.

#### radiusMatch public void radiusMatch([Mat](http://docs.google.com/org/opencv/core/Mat.html) queryDescriptors, [Mat](http://docs.google.com/org/opencv/core/Mat.html) trainDescriptors, java.util.List<[MatOfDMatch](http://docs.google.com/org/opencv/core/MatOfDMatch.html)> matches, float maxDistance, [Mat](http://docs.google.com/org/opencv/core/Mat.html) mask, boolean compactResult) For each query descriptor, finds the training descriptors not farther than the specified distance.Parameters:queryDescriptors - Query set of descriptors.trainDescriptors - Train set of descriptors. This set is not added to the train descriptors collection stored in the class object.matches - Found matches.compactResult - Parameter used when the mask (or masks) is not empty. If compactResult is false, the matches vector has the same size as queryDescriptors rows. If compactResult is true, the matches vector does not contain matches for fully masked-out query descriptors.maxDistance - Threshold for the distance between matched descriptors. Distance means here metric distance (e.g. Hamming distance), not the distance between coordinates (which is measured in Pixels)!mask - Mask specifying permissible matches between an input query and train matrices of descriptors. For each query descriptor, the methods find such training descriptors that the distance between the query descriptor and the training descriptor is equal or smaller than maxDistance. Found matches are returned in the distance increasing order.

#### read public void read(java.lang.String fileName)

#### train public void train() Trains a descriptor matcher Trains a descriptor matcher (for example, the flann index). In all methods to match, the method train() is run every time before matching. Some descriptor matchers (for example, BruteForceMatcher) have an empty implementation of this method. Other matchers really train their inner structures (for example, FlannBasedMatcher trains flann::Index ).

#### write public void write(java.lang.String fileName)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Tree](http://docs.google.com/package-tree.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* Prev Class
* [Next Class](http://docs.google.com/org/opencv/features2d/FastFeatureDetector.html)
* [Frames](http://docs.google.com/index.html?org/opencv/features2d/DescriptorMatcher.html)
* [No Frames](http://docs.google.com/DescriptorMatcher.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Field](#3znysh7) |
* Constr |
* [Method](#2et92p0)
* Detail:
* [Field](#1t3h5sf) |
* Constr |
* [Method](#35nkun2)

Generated on 2021-04-02 03:15:03 / OpenCV 3.4.14