com._604robotics.robot2012.vision.config

Class LinkedSlider.DoubleLinkedSlider

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.Box
          com._604robotics.robot2012.vision.config.LinkedSlider
            com._604robotics.robot2012.vision.config.LinkedSlider.DoubleLinkedSlider

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.event.ChangeListener

Direct Known Subclasses:

LinkedSlider.ExponentialLinkedSlider

Enclosing class:

LinkedSlider

```
public static class LinkedSlider.DoubleLinkedSlider
extends LinkedSlider
```

A LinkedSlider that can be set to floating-point values

See Also:

Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

LinkedSlider.DoubleLinkedSlider, LinkedSlider.ExponentialLinkedSlider, LinkedSlider.IntLinkedSlider

Nested classes/interfaces inherited from class javax.swing.Box

javax.swing.Box.AccessibleBox, javax.swing.Box.Filler

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

Nested classes/interfaces inherited from class java.awt.Container

java.awt.Container.AccessibleAWTContainer

Nested classes/interfaces inherited from class java.awt.Component

java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy

Field Summary

Fields inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

max, min, mul, slider

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

Constructors

Constructor and Description
LinkedSlider.DoubleLinkedSlider(java.lang.String name, double initialValue, double max) A constructor for a DoubleLinkedSlider

Method Summary

Methods

Modifier and Type	Method and Description
double	**getValue**()
void	**setValue**(double val) A setter for the value of the slider

Methods inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

getValText, stateChanged, updateValLabel

Methods inherited from class javax.swing.Box

createGlue, createHorizontalBox, createHorizontalGlue, createHorizontalStrut, createRigidArea, createVerticalBox, createVerticalGlue, createVerticalStrut, getAccessibleContext, paintComponent, setLayout

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage

checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

LinkedSlider.DoubleLinkedSlider

```
public LinkedSlider.DoubleLinkedSlider(java.lang.String name,
                                       double initialValue,
                                       double max)
```

A constructor for a DoubleLinkedSlider

Parameters:

`name` - The name of the slider

`initialValue` - The initial value

`max` - The maximum value that this slider can be at

Method Detail

getValue

```
public double getValue()
```

Specified by:

`getValue` in class `LinkedSlider`

Returns:

The current value

setValue

```
public void setValue(double val)
```

Description copied from class: `LinkedSlider`
A setter for the value of the slider

Specified by:

`setValue` in class `LinkedSlider`

Parameters:

`val` - the value to set the slider to

com._604robotics.robot2012.vision.config

Class LinkedSlider

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.Box
          com._604robotics.robot2012.vision.config.LinkedSlider

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.event.ChangeListener

Direct Known Subclasses:

LinkedSlider.DoubleLinkedSlider, LinkedSlider.IntLinkedSlider

---

```
public abstract class LinkedSlider
extends javax.swing.Box
implements javax.swing.event.ChangeListener
```

A JSlider that displays its current position and name in JLabels next to it

See Also:

Serialized Form

Nested Class Summary

Nested Classes

Modifier and Type	Class and Description
static class	**LinkedSlider.DoubleLinkedSlider** A LinkedSlider that can be set to floating-point values
static class	**LinkedSlider.ExponentialLinkedSlider** A LinkedSlider that has an exponential scale, so it is much easier to pick small values (close to zero) while still allowing a range up to 1
static class	**LinkedSlider.IntLinkedSlider** A LinkedSlider that can only be set to integers

Nested classes/interfaces inherited from class javax.swing.Box

javax.swing.Box.AccessibleBox, javax.swing.Box.Filler

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

Nested classes/interfaces inherited from class java.awt.Container

java.awt.Container.AccessibleAWTContainer

Nested classes/interfaces inherited from class java.awt.Component

java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy

Field Summary

Fields

Modifier and Type	Field and Description
protected int	**max** The maximum value on the slider (must be an integer)
protected int	**min** The minimum value on the slider (must be an integer)

protected double	**mul** A number to multiply all slider outputs by
javax.swing.JSlider	**slider** The slider that the user interacts with

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

Constructors

Constructor and Description
LinkedSlider(java.lang.String name, int min, int max, int val) A constructor for a LinkedSlider

Method Summary

Methods

Modifier and Type	Method and Description
java.lang.String	**getValText**() This method returns a human-readable formatted number suited for the type of LinkedSlider.
abstract double	**getValue**()
abstract void	**setValue**(double val) A setter for the value of the slider
void	**stateChanged**(javax.swing.event.ChangeEvent e)
protected void	**updateValLabel**() This method updates the label on the right side that displays the current value

Methods inherited from class javax.swing.Box

createGlue, createHorizontalBox, createHorizontalGlue, createHorizontalStrut, createRigidArea, createVerticalBox, createVerticalGlue, createVerticalStrut, getAccessibleContext, paintComponent, setLayout

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener,

applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

slider

```
public javax.swing.JSlider slider
```

The slider that the user interacts with

min

```
protected int min
```

The minimum value on the slider (must be an integer)

max

```
protected int max
```

The maximum value on the slider (must be an integer)

mul

```
protected double mul
```

A number to multiply all slider outputs by

Constructor Detail

LinkedSlider

```
public LinkedSlider(java.lang.String name,
            int min,
            int max,
            int val)
```

A constructor for a LinkedSlider

Parameters:

`name` - The name of the slider

`min` - The minimum value

`max` - The maximum value

`val` - The initial value

Method Detail

setValue

```
public abstract void setValue(double val)
```

A setter for the value of the slider

Parameters:

`val` - the value to set the slider to

stateChanged

```
public void stateChanged(javax.swing.event.ChangeEvent e)
```

Specified by:

`stateChanged` in interface `javax.swing.event.ChangeListener`

updateValLabel

```
protected void updateValLabel()
```

This method updates the label on the right side that displays the current value

getValText

```
public java.lang.String getValText()
```

This method returns a human-readable formatted number suited for the type of LinkedSlider. It is used to show the current value on the slider

Returns:

The string that is shown in the JLabel to the right of the slider

getValue

```
public abstract double getValue()
```

Returns:

The current value

com._604robotics.robot2012.vision.config

Class Configger

java.lang.Object
    com._604robotics.robot2012.vision.config.Configger

```
public class Configger
extends java.lang.Object
```

This class creates a window for configuring various aspects of the Vision program, such as target color, target sensitivity, and other values found in `Config`.

The name of this class is officially "Configger", a common mispronunciation of the word "Configure". It comes from nounifying the verb form of the shortened word "Config".

Constructor Summary

Constructors

Constructor and Description
`Configger``()` This constructor of the Configger initializes everything and sets the Configger as visible.

Method Summary

Methods

Modifier and Type	Method and Description
`static javax.swing.Box`	**`boxForTextField`**`(javax.swing.JTextField textField, java.lang.String name)` A simple utility method that creates a javax.swing.Box that holds a label indicating the name of the variable to change and a text field for the user to type input into.
`static void`	**`main`**`(java.lang.String[] args)` A simple main() method to make the Configger a runnable program

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

Configger

```
public Configger()
```

This constructor of the Configger initializes everything and sets the Configger as visible.

Method Detail

main

```
public static void main(java.lang.String[] args)
```

A simple main() method to make the Configger a runnable program

boxForTextField

```
public static javax.swing.Box boxForTextField(javax.swing.JTextField textField,
                                              java.lang.String name)
```

A simple utility method that creates a javax.swing.Box that holds a label indicating the name of the variable to change and a text field for the user to type input into.

Parameters:

`textField` - The JTextField the user can type into

`name` - The name of the value to change (shown in a JLabel)

Returns:

a Box containing the JLabel and JTextField

com._604robotics.robot2012.vision.config

Class Config

java.lang.Object
    com._604robotics.robot2012.vision.config.Config

```
public class Config
extends java.lang.Object
```

The configuration of the Team 604 FRCVision

Field Summary

Fields

Modifier and Type	Field and Description
boolean	**checkCenter** Should the tiling algorithm check the center of the tile, as well as the corners to determine if it should be considered for being in the target?
double	**color_mulB** How much to multiply the square of the errors per color channel by
double	**color_mulG** How much to multiply the square of the errors per color channel by
double	**color_mulR** How much to multiply the square of the errors per color channel by
int	**color_targetB** The color of the vision target when the light is shining on it
int	**color_targetG** The color of the vision target when the light is shining on it
int	**color_targetR** The color of the vision target when the light is shining on it
boolean	**communicateToRobot** Should this program attempt to communicate to the robot?
boolean	**debug_Print** Should debug info be shown? This includes time per frame, number of visible targets, and estimated position of visible targets.
boolean	**debug_SaveImagesToFiles** Should camera images be stored onto disk, for debug purposes?
boolean	**debug_ShowDisplay** Should the fancy display be shown, with green and red tiles indicating matching and non-matching tiles, with blue lines and dots indicating target sides and corners?
int	**minBlobSize** A calibration constant indicating the minimum size for a potential target to be considered.
boolean	**scanWholeTile** Should all pixels be scanned in every tile be scanned, or just the corners (and possibly center)
byte	**sensitivity** A constant between -128 to +127 indicating how sensitive the color acceptance of the target should be.
int	**tileSize** The size of each tile in the vision processing.

Constructor Summary

Constructors

Constructor and Description
Config()

Method Summary

Methods

Modifier and Type	Method and Description

static Config	readConfig(java.io.File file) Read a Config from a file
static Config	readDefaultConfig() Reads the default Config file
void	save(java.io.File file) Saves this Config to a given file
void	saveDefaultConfig() Saves this Config to the default file
java.lang.String	toString()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

checkCenter

```
public boolean checkCenter
```

Should the tiling algorithm check the center of the tile, as well as the corners to determine if it should be considered for being in the target?

communicateToRobot

```
public boolean communicateToRobot
```

Should this program attempt to communicate to the robot?

debug_Print

```
public boolean debug_Print
```

Should debug info be shown? This includes time per frame, number of visible targets, and estimated position of visible targets.

debug_SaveImagesToFiles

```
public boolean debug_SaveImagesToFiles
```

Should camera images be stored onto disk, for debug purposes?

debug_ShowDisplay

```
public boolean debug_ShowDisplay
```

Should the fancy display be shown, with green and red tiles indicating matching and non-matching tiles, with blue lines and dots indicating target sides and corners?

minBlobSize

```
public int minBlobSize
```

A calibration constant indicating the minimum size for a potential target to be considered. This number is given in square "tiles", with `tileSize` pixels side lengths

scanWholeTile

```
public boolean scanWholeTile
```

Should all pixels be scanned in every tile be scanned, or just the corners (and possibly center)

sensitivity

```
public byte sensitivity
```

A constant between -128 to +127 indicating how sensitive the color acceptance of the target should be. Lower numbers will allow more pixels, while higher numbers will eliminate more.
This number needs to be chosen high enough to reduce or eliminate false positives, but it needs to be low enough to not generate false negatives.

tileSize

tileSize

```
public int tileSize
```

The size of each tile in the vision processing. This is represented in pixels. It should be a number chosen large enough to have a good speed, but small enough to not miss a target in the image.

color_targetR

```
public int color_targetR
```

The color of the vision target when the light is shining on it

color_targetG

```
public int color_targetG
```

The color of the vision target when the light is shining on it

color_targetB

```
public int color_targetB
```

The color of the vision target when the light is shining on it

color_mulR

```
public double color_mulR
```

How much to multiply the square of the errors per color channel by

color_mulG

```
public double color_mulG
```

How much to multiply the square of the errors per color channel by

color_mulB

```
public double color_mulB
```

How much to multiply the square of the errors per color channel by

Constructor Detail

Config

```
public Config()
```

Method Detail

readDefaultConfig

```
public static Config readDefaultConfig()
```

Reads the default Config file

Returns:

the Config, as read from vision.conf

saveDefaultConfig

```
public void saveDefaultConfig()
                     throws java.io.IOException
```

Saves this Config to the default file

Throws:

java.io.IOException - If an error occurs

java.io.IOException - If an error occurs

readConfig

```
public static Config readConfig(java.io.File file)
```

Read a Config from a file

Parameters:

`file` - the file to read it from

Returns:

the Config read from the file

save

```
public void save(java.io.File file)
          throws java.io.IOException
```

Saves this Config to a given file

Parameters:

`file` - The file to save to

Throws:

`java.io.IOException` - If an error occurs

toString

```
public java.lang.String toString()
```

Overrides:

`toString` in class `java.lang.Object`

com._604robotics.robot2012.vision.config

Class LinkedSlider.IntLinkedSlider

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.Box
          com._604robotics.robot2012.vision.config.LinkedSlider
            com._604robotics.robot2012.vision.config.LinkedSlider.IntLinkedSlider

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.event.ChangeListener

Enclosing class:

LinkedSlider

```
public static class LinkedSlider.IntLinkedSlider
extends LinkedSlider
```

A LinkedSlider that can only be set to integers

See Also:

Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

LinkedSlider.DoubleLinkedSlider, LinkedSlider.ExponentialLinkedSlider, LinkedSlider.IntLinkedSlider

Nested classes/interfaces inherited from class javax.swing.Box

javax.swing.Box.AccessibleBox, javax.swing.Box.Filler

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

Nested classes/interfaces inherited from class java.awt.Container

java.awt.Container.AccessibleAWTContainer

Nested classes/interfaces inherited from class java.awt.Component

java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy

Field Summary

Fields inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

max, min, mul, slider

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

Constructors

Constructor and Description
LinkedSlider.IntLinkedSlider(java.lang.String name, int min, int max, int val) A constructor

Method Summary

Methods

Modifier and Type	Method and Description
int	**getIntValue**()
java.lang.String	**getValText**() This method returns a human-readable formatted number suited for the type of LinkedSlider.
double	**getValue**()
void	**setValue**(double val) A setter for the value of the slider

Methods inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

stateChanged, updateValLabel

Methods inherited from class javax.swing.Box

createGlue, createHorizontalBox, createHorizontalGlue, createHorizontalStrut, createRigidArea, createVerticalBox, createVerticalGlue, createVerticalStrut, getAccessibleContext, paintComponent, setLayout

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds,

checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

LinkedSlider.IntLinkedSlider

```
public LinkedSlider.IntLinkedSlider(java.lang.String name,
                            int min,
                            int max,
                            int val)
```

A constructor

Parameters:

`name` - The name of the slider

`min` - The minimum value

`max` - The maximum value

`val` - The initial value

Method Detail

getIntValue

```
public int getIntValue()
```

Returns:

the current value

getValue

```
public double getValue()
```

Specified by:

`getValue` in class `LinkedSlider`

Returns:

The current value

getValText

```
public java.lang.String getValText()
```

Description copied from class: `LinkedSlider`

This method returns a human-readable formatted number suited for the type of LinkedSlider. It is used to show the current value on the slider

Overrides:

`getValText` in class `LinkedSlider`

Returns:

The string that is shown in the JLabel to the right of the slider

setValue

```
public void setValue(double val)
```

Description copied from class: `LinkedSlider`
A setter for the value of the slider

Specified by:

`setValue` in class `LinkedSlider`

Parameters:

`val` - the value to set the slider to

com._604robotics.robot2012.vision.config

Class LinkedSlider.ExponentialLinkedSlider

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.Box
          com._604robotics.robot2012.vision.config.LinkedSlider
            com._604robotics.robot2012.vision.config.LinkedSlider.DoubleLinkedSlider
              com._604robotics.robot2012.vision.config.LinkedSlider.ExponentialLinkedSlider

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.event.ChangeListener

Enclosing class:

LinkedSlider

```
public static class LinkedSlider.ExponentialLinkedSlider
extends LinkedSlider.DoubleLinkedSlider
```

A LinkedSlider that has an exponential scale, so it is much easier to pick small values (close to zero) while still allowing a range up to 1

See Also:

Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

LinkedSlider.DoubleLinkedSlider, LinkedSlider.ExponentialLinkedSlider, LinkedSlider.IntLinkedSlider

Nested classes/interfaces inherited from class javax.swing.Box

javax.swing.Box.AccessibleBox, javax.swing.Box.Filler

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

Nested classes/interfaces inherited from class java.awt.Container

java.awt.Container.AccessibleAWTContainer

Nested classes/interfaces inherited from class java.awt.Component

java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy

Field Summary

Fields inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

max, min, mul, slider

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM ALIGNMENT, CENTER ALIGNMENT, LEFT ALIGNMENT, RIGHT ALIGNMENT, TOP ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

Constructors

Constructor and Description
LinkedSlider.ExponentialLinkedSlider(java.lang.String name, double initial) A constructor to make an ExponentialLinkedSlider.
LinkedSlider.ExponentialLinkedSlider(java.lang.String name, double initial, double max) A constructor to make an ExponentialLinkedSlider

Method Summary

Methods

Modifier and Type	Method and Description
double	**getValue**()
void	**setValue**(double val) A setter for the value of the slider

Methods inherited from class com._604robotics.robot2012.vision.config.LinkedSlider

getValText, stateChanged, updateValLabel

Methods inherited from class javax.swing.Box

createGlue, createHorizontalBox, createHorizontalGlue, createHorizontalStrut, createRigidArea, createVerticalBox, createVerticalGlue, createVerticalStrut, getAccessibleContext, paintComponent, setLayout

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage

checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

LinkedSlider.ExponentialLinkedSlider

```
public LinkedSlider.ExponentialLinkedSlider(java.lang.String name,
                                            double initial,
                                            double max)
```

A constructor to make an ExponentialLinkedSlider

Parameters:

`name` - The name of the slider

`initial` - The initial value

`max` - The maximum value

LinkedSlider.ExponentialLinkedSlider

```
public LinkedSlider.ExponentialLinkedSlider(java.lang.String name,
                                            double initial)
```

A constructor to make an ExponentialLinkedSlider. A default max of 1 is assumed.

Parameters:

`name` - The name of the slider

`initial` - the initial value of the slider

Method Detail

getValue

```
public double getValue()
```

Overrides:

`getValue` in class `LinkedSlider.DoubleLinkedSlider`

Returns:

The current value

setValue

```
public void setValue(double val)
```

Description copied from class: `LinkedSlider`

A setter for the value of the slider

Overrides:

`setValue` in class `LinkedSlider.DoubleLinkedSlider`

Parameters:

`val` - the value to set the slider to

com._604robotics.robot2012.vision

Class LinearRegression.RegressionResult

java.lang.Object
    com._604robotics.robot2012.vision.LinearRegression.RegressionResult

Direct Known Subclasses:

LinearRegression.BackwardsRegressionResult

Enclosing class:

LinearRegression

```
public static class LinearRegression.RegressionResult
extends java.lang.Object
```

A regression result that indicates the line that best matches a given set of data.

Constructor Summary

Constructors

Constructor and Description
`LinearRegression.RegressionResult(double m, double b, double r2)`

Method Summary

Methods

Modifier and Type	Method and Description
`java.lang.String`	`toString()`

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait`

Constructor Detail

LinearRegression.RegressionResult

```
public LinearRegression.RegressionResult(double m,
                                         double b,
                                         double r2)
```

Parameters:

`m` - The slope of the regression line

`b` - The y-intercept of the regression line

`r2` - A number indicating how good of a fit this line is

Method Detail

toString

```
public java.lang.String toString()
```

Overrides:

`toString` in class `java.lang.Object`

com._604robotics.robot2012.vision

Class Point3d

java.lang.Object
    com._604robotics.robot2012.vision.Point3d

```
public class Point3d
extends java.lang.Object
```

This represents a point in 3d space

Field Summary

Fields

Modifier and Type	Field and Description
double	**x** The X value
double	**y** The Y value
double	**z** The Z value

Constructor Summary

Constructors

Constructor and Description
Point3d(double x, double y, double z)

Method Summary

Methods

Modifier and Type	Method and Description
double	**getX**()
double	**getY**()
double	**getZ**()
void	**setX**(double x) Sets the X value of this Point
void	**setY**(double y) Sets the Y value of this Point
void	**setZ**(double z) Sets the Z value of this Point

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

x

```
public double x
```

The X value

y

y

```
public double y
```

The Y value

z

```
public double z
```

The Z value

Constructor Detail

Point3d

```
public Point3d(double x,
       double y,
       double z)
```

Parameters:

`x` - The X value

`y` - The Y value

`z` - The Z value

Method Detail

getX

```
public double getX()
```

Returns:

The X value

setX

```
public void setX(double x)
```

Sets the X value of this Point

Parameters:

`x` - The X value

getY

```
public double getY()
```

Returns:

The Y value

setY

```
public void setY(double y)
```

Sets the Y value of this Point

Parameters:

`y` - The Y value

getZ

```
public double getZ()
```

Returns:

The Z value

setZ

```
public void setZ(double z)
```

Sets the Z value of this Point

Parameters:

`z` - The Z value

com._604robotics.robot2012.vision

Class LinearRegression.BackwardsRegressionResult

java.lang.Object
  com._604robotics.robot2012.vision.LinearRegression.RegressionResult
    com._604robotics.robot2012.vision.LinearRegression.BackwardsRegressionResult

Enclosing class:

LinearRegression

```
public static class LinearRegression.BackwardsRegressionResult
extends LinearRegression.RegressionResult
```

A regression result that, instead of having y as a function of x has x as a function of y.

See Also:

LinearRegression.BackwardsRegressionResult

Constructor Summary

Constructors

Constructor and Description
LinearRegression.BackwardsRegressionResult(double m, double b, double r2)

Method Summary

Methods inherited from class com._604robotics.robot2012.vision.LinearRegression.RegressionResult

toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

LinearRegression.BackwardsRegressionResult

```
public LinearRegression.BackwardsRegressionResult(double m,
                                                  double b,
                                                  double r2)
```

com._604robotics.robot2012.vision

Class Point2d

java.lang.Object
    com._604robotics.robot2012.vision.Point2d

```
public class Point2d
extends java.lang.Object
```

This represents a Point in 2d space

Field Summary

Fields

Modifier and Type	Field and Description
double	**x** The X value
double	**y** The Y value

Constructor Summary

Constructors

Constructor and Description
Point2d(double x, double y)

Method Summary

Methods

Modifier and Type	Method and Description
double	**getX**()
double	**getY**()
void	**setX**(double x) Sets the X value of this Point
void	**setY**(double y) Sets the Y value of this Point
java.lang.String	**toString**()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

x

```
public double x
```

The X value

y

```
public double y
```

The Y value

Constructor Detail

Point2d

```
public Point2d(double x,
       double y)
```

Parameters:

`x` - the X value

`y` - the Y value

Method Detail

getX

```
public double getX()
```

Returns:

the X value

getY

```
public double getY()
```

Returns:

the Y value

setX

```
public void setX(double x)
```

Sets the X value of this Point

Parameters:

`x` - the X value

setY

```
public void setY(double y)
```

Sets the Y value of this Point

Parameters:

`y` - the Y value

toString

```
public java.lang.String toString()
```

Overrides:

`toString` in class `java.lang.Object`

com._604robotics.robot2012.vision

Class Target

java.lang.Object
    com._604robotics.robot2012.vision.Target

All Implemented Interfaces:

java.lang.Comparable<Target>

```
public class Target
extends java.lang.Object
implements java.lang.Comparable<Target>
```

This class represents a physical vision Target with four main attributes (x, y, z, angle). As well, there are estimated uncertainties attached to all of these numbers.

To get the position of the hoop, use the DistanceCalculations class.

Field Summary

Fields

Modifier and Type	Field and Description
double	**angle** This is the angle of the target, relative to the camera.
double	**angleUncertainty** This is the uncertainty of the angle of the target.
static double	**RelHoopY** The distance from the center of the target to the Y (vertical) value of the hoop.
static double	**RelHoopZ** The distance from the center of the target to the Z (depth) value of the hoop.
double	**x** x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.
double	**xUncertainty** These are the uncertainties of the x, y, and z positions of the target.
double	**y** x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.
double	**yUncertainty** These are the uncertainties of the x, y, and z positions of the target.
double	**z** x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.
double	**zUncertainty** These are the uncertainties of the x, y, and z positions of the target.

Constructor Summary

Constructors

Constructor and Description
Target() A blank constructor to easily make a Target
Target(double x, double y, double z, double angle)
Target(double x, double y, double z, double xUncertainty, double yUncertainty, double zUncertainty, double angle, double angleUncertainty)
Target(**Point3d** point, double angle)

Method Summary

Methods

Modifier and Type	Method and Description
int	**compareTo**(**Target** that)
double	**getAngle**()
double	**getAngleUncertainty**()
Point3d	**getHoopPosition**()
Point3d	**getReflectedHoopPosition**()
Point3d	**getReflectedHoopPosition**(double bounceFactor)
double	**getX**()
double	**getXUncertainty**()
double	**getY**()
double	**getYUncertainty**()
double	**getZ**()
double	**getZUncertainty**()
void	**setAngle**(double angle)
void	**setAngleUncertainty**(double angleUncertainty)
void	**setPoint**(**Point3d** point)
void	**setX**(double x)
void	**setXUncertainty**(double xUncertainty)
void	**setY**(double y)
void	**setYUncertainty**(double yUncertainty)
void	**setZ**(double z)
void	**setZUncertainty**(double zUncertainty)
java.lang.String	**toString**()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

RelHoopY

```
public static final double RelHoopY
```

The distance from the center of the target to the Y (vertical) value of the hoop.

See Also:

Constant Field Values

RelHoopZ

```
public static final double RelHoopZ
```

The distance from the center of the target to the Z (depth) value of the hoop.

See Also:

Constant Field Values

angle

```
public double angle
```

This is the angle of the target, relative to the camera.

(angle)
.....(Target)
...../
...../
...../
..../ - - - - - - - |> (Camera)
.../
../
./
/
this value is expressed in radians.

angleUncertainty

angleUncertainty

```
public double angleUncertainty
```

This is the uncertainty of the angle of the target. This is interpreted as a plus or minus to the angle. Again, this is expressed in radians

x

```
public double x
```

x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera. y will be positive when the target appears to be above of the center of the camera. z will always be negative (see Wikipedia: Right-hand rule). As the absolute value of z increases, so does the distance from the camera to the target. To determine the approximate accuracy of these values, check [x, y, z]_accuracy. The units of these measures are in inches.

y

```
public double y
```

x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera. y will be positive when the target appears to be above of the center of the camera. z will always be negative (see Wikipedia: Right-hand rule). As the absolute value of z increases, so does the distance from the camera to the target. To determine the approximate accuracy of these values, check [x, y, z]_accuracy. The units of these measures are in inches.

z

```
public double z
```

x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera. y will be positive when the target appears to be above of the center of the camera. z will always be negative (see Wikipedia: Right-hand rule). As the absolute value of z increases, so does the distance from the camera to the target. To determine the approximate accuracy of these values, check [x, y, z]_accuracy. The units of these measures are in inches.

xUncertainty

```
public double xUncertainty
```

These are the uncertainties of the x, y, and z positions of the target. These are interpreted as pluses and minuses to the x, y, and z values. Again, these are in inches.

yUncertainty

```
public double yUncertainty
```

These are the uncertainties of the x, y, and z positions of the target. These are interpreted as pluses and minuses to the x, y, and z values. Again, these are in inches.

zUncertainty

```
public double zUncertainty
```

These are the uncertainties of the x, y, and z positions of the target. These are interpreted as pluses and minuses to the x, y, and z values. Again, these are in inches.

Constructor Detail

Target

```
public Target()
```

A blank constructor to easily make a Target

Target

```
public Target(double x,
      double y,
      double z,
      double angle)
```

Parameters:

`x` - the X coordinate of the center of the vision target

`y` - the Y coordinate of the center of the vision target

`z` - the Z coordinate of the center of the vision target

`angle` -

Target

```
public Target(double x,
      double y,
      double z,
      double xUncertainty,
      double yUncertainty,
      double zUncertainty,
      double angle,
      double angleUncertainty)
```

Parameters:

`x` - the X coordinate of the center of the vision target

`y` - the Y coordinate of the center of the vision target

`z` - the Z coordinate of the center of the vision target

`xUncertainty` - the X Uncertainty

`yUncertainty` - the Y Uncertainty

`zUncertainty` - the Z Uncertainty

`angle` - the Angle

`angleUncertainty` - the Angle Uncertainty

Target

```
public Target(Point3d point,
      double angle)
```

Parameters:

`point` - the Point

`angle` - the Angle

Method Detail

compareTo

```
public int compareTo(Target that)
```

Specified by:

`compareTo` in interface `java.lang.Comparable<Target>`

getAngle

```
public double getAngle()
```

Returns:

the angle that the vision target faces

getAngleUncertainty

```
public double getAngleUncertainty()
```

Returns:

the uncertainty of the Angle

getHoopPosition

```
public Point3d getHoopPosition()
```

Returns:

the position of the hoop accounting for the fact that the center of the hoop is not at the center of the target

getReflectedHoopPosition

```
public Point3d getReflectedHoopPosition()
```

Returns:

the reflected position of the hoop accounting for the fact that the center of the hoop is not at the center of the target. This is useful bounces

getReflectedHoopPosition

```
public Point3d getReflectedHoopPosition(double bounceFactor)
```

Parameters:

`bounceFactor` - a number that scales the changes in the x and z distances due to correction for hoop position. In a idealized collision, this is equal to the inverse of its coefficient of restitution. However, with spin, this number should be less.

Returns:

the reflected position of the hoop accounting for the fact that the center of the hoop is not at the center of the target. This is useful bounces

getX

```
public double getX()
```

Returns:

the X coordinate of the center of the vision target

getXUncertainty

```
public double getXUncertainty()
```

Returns:

the Uncertainty of the X coordinate

getY

```
public double getY()
```

Returns:

the Y coordinate of the center of the vision target

getYUncertainty

```
public double getYUncertainty()
```

Returns:

the Uncertainty of the Y coordinate

getZ

```
public double getZ()
```

Returns:

the Z coordinate of the center of the vision target

getZUncertainty

```
public double getZUncertainty()
```

Returns:

the Uncertainty of the Z coordinate of the vision target

setAngle

```
public void setAngle(double angle)
```

Parameters:

`angle` - the Angle to set

setAngleUncertainty

```
public void setAngleUncertainty(double angleUncertainty)
```

Parameters:

`angleUncertainty` - the angleUncertainty to set

setPoint

```
public void setPoint(Point3d point)
```

Parameters:

`point` - the point to set the center of this target

setX

```
public void setX(double x)
```

Parameters:

`x` - the X to set

setXUncertainty

```
public void setXUncertainty(double xUncertainty)
```

Parameters:

`xUncertainty` - the xUncertainty to set

setY

```
public void setY(double y)
```

Parameters:

`y` - the Y to set

setYUncertainty

```
public void setYUncertainty(double yUncertainty)
```

Parameters:

`yUncertainty` - the yUncertainty to set

setZ

```
public void setZ(double z)
```

Parameters:

`z` - the Z to set

setZUncertainty

```
public void setZUncertainty(double zUncertainty)
```

Parameters:

`zUncertainty` - the zUncertainty to set

toString

```
public java.lang.String toString()
```

public java.lang.String toString()

Overrides:

`toString` in class `java.lang.Object`

com._604robotics.robot2012.vision

Class VisionProcessing

java.lang.Object
    com._604robotics.robot2012.vision.VisionProcessing

```
public class VisionProcessing
extends java.lang.Object
```

The main class for processing camera vision on our 2012 robot. This software takes in camera images from the robot's camera, parses them, searches for pixels that look like shiny blue vision targets, blobs those pixels together, (if they are connected), and then treats it as a quadrilateral and finds the corners.

Field Summary

Fields

Modifier and Type	Field and Description
`Config`	**conf** The Configuration file for this VisionProcessing
`static VisionProcessing`	**defaultProcessing** The default VisionProcessing to use; this should be where the root of all of the vision processing is done
`VisionDisp`	**display** The display for showing the image as well as some debug data.
`static int`	**Side_Bottom** Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.
`static int`	**Side_Left** Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.
`static int`	**Side_Right** Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.
`static int`	**Side_Top** Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.

Constructor Summary

Constructors

Constructor and Description
VisionProcessing() A constructor to create a new VisionProcessing

Method Summary

Methods

Modifier and Type	Method and Description
`LinearRegression.RegressionResult`	**getRegressionForSide**(`ResultImage ri, int side, AABB guess`) Get a line that best fits the sides of a given target
`void`	**loopAndProcessPics**() This function waits for images from the image stream, processes them, and then sends results to the robot.
`void`	**loopAndProcessPreSavedPics**() This function is just a simple debug function for testing with pre-saved images.
`static void`	**main**(`java.lang.String[] args`) Just a simple main() function for running and testing the target tracking
`void`	**processImage**(`java.awt.image.BufferedImage img`) This processes the camera image and can send it to the robot (if enabled in the config file)
`static void`	**recursiveTraceBlobs**(`Img results, int i, int j, int color`)

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Field Detail

defaultProcessing

```
public static final VisionProcessing defaultProcessing
```

The default VisionProcessing to use; this should be where the root of all of the vision processing is done

Side_Left

```
public static final int Side_Left
```

Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.

See Also:

Constant Field Values

Side_Top

```
public static final int Side_Top
```

Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.

See Also:

Constant Field Values

Side_Right

```
public static final int Side_Right
```

Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.

See Also:

Constant Field Values

Side_Bottom

```
public static final int Side_Bottom
```

Constants indicating the Left, Top, Right, and Bottom sides of a target or bounding box.

See Also:

Constant Field Values

conf

```
public Config conf
```

The Configuration file for this VisionProcessing

display

```
public final VisionDisp display
```

The display for showing the image as well as some debug data. It shows targets in green, and sides and corners in blue.

Constructor Detail

VisionProcessing

```
public VisionProcessing()
```

A constructor to create a new VisionProcessing

Method Detail

getRegressionForSide

```
public LinearRegression.RegressionResult getRegressionForSide(ResultImage ri,
                                                               int side,
                                                               AABB guess)
```

Get a line that best fits the sides of a given target

Parameters:

`ri` - the ResultImage that indicates which pixels are contained in the target

`side` - an integer indicating which of the sides to pick

`guess` - a bounding box that surrounds all of the pixels to check

Returns:

the line of best fit for the given side of the target lying in the AABB

main

```
public static void main(java.lang.String[] args)
                 throws java.lang.InterruptedException,
                        java.io.IOException
```

Just a simple main() function for running and testing the target tracking

Throws:

`java.lang.InterruptedException`

`java.io.IOException`

recursiveTraceBlobs

```
public static void recursiveTraceBlobs(Img results,
                       int i,
                       int j,
                       int color)
```

Parameters:

`results` - the Img to store returned data in

`i` - the X coordinate

`j` - the Y coordinate

`color` - the blob's color

loopAndProcessPics

```
public void loopAndProcessPics()
                        throws java.net.MalformedURLException
```

This function waits for images from the image stream, processes them, and then sends results to the robot.

Throws:

`java.net.MalformedURLException`

loopAndProcessPreSavedPics

```
public void loopAndProcessPreSavedPics()
                                 throws java.io.IOException
```

This function is just a simple debug function for testing with pre-saved images. Currently, it just reads over a loop of 50 pictures saved as target/[number].jpeg

Throws:

`java.io.IOException`

processImage

```
public void processImage(java.awt.image.BufferedImage img)
```

This processes the camera image and can send it to the robot (if enabled in the config file)

Parameters:

`img` - an image as received from the camera

com._604robotics.robot2012.vision

Class Result.PlusResult

java.lang.Object
  com._604robotics.robot2012.vision.Result
    com._604robotics.robot2012.vision.Result.PlusResult

Enclosing class:

Result

```
public static class Result.PlusResult
extends Result
```

A result indicating that it is likely that the target lies in the indicated tile

Nested Class Summary

Nested classes/interfaces inherited from class com._604robotics.robot2012.vision.Result

Result.AntiResult, Result.PlusResult

Constructor Summary

Constructors

Constructor and Description
Result.PlusResult(int tileSize, byte[] dat) A simple constructor to make a PlusResult.

Method Summary

Methods

Modifier and Type	Method and Description
boolean	**hasPlus**()
boolean	**plusAt**(int x, int y)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Result.PlusResult

```
public Result.PlusResult(int tileSize,
                         byte[] dat)
```

A simple constructor to make a PlusResult.

Parameters:

`tileSize` - the size of this tile

`dat` - the array of bytes indicating how well the pixel matches the target.

Method Detail

hasPlus

```
public boolean hasPlus()
```

Overrides:

`hasPlus` in class `Result`

Returns:

whether there are any pixels matching the color of the target or not

plusAt

```
public boolean plusAt(int x,
                      int y)
```

Overrides:

`plusAt` in class `Result`

Parameters:

`x` - the X coordinate (within the tile, not the image)

`y` - the Y coordinate (within the tile, not the image)

Returns:

whether or not the pixel at the given location matches the Target color

com._604robotics.robot2012.vision

Class ResultImage

java.lang.Object
    com._604robotics.robot2012.vision.ResultImage

```
public class ResultImage
extends java.lang.Object
```

A result image that holds an image of how well pixels match the expected color of the vision target. It is treated like a giant boolean array externally, but internally it is split up into small tiles.

See Also:

Result

Field Summary

Fields

Modifier and Type	Field and Description
Result[]	results

Constructor Summary

Constructors

Constructor and Description
ResultImage(int imW, int imH) A constructor to create a new ResultImage.

Method Summary

Methods

Modifier and Type	Method and Description
void	computeResults(Img img) This method goes through an Img and finds which pixels appear to match the color of the vision target.
boolean	isTarget(int x, int y)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

results

```
public Result[] results
```

Constructor Detail

ResultImage

```
public ResultImage(int imW,
           int imH)
```

A constructor to create a new ResultImage. To actually initialize the returned ResultImage, use ResultImage

A constructor to create a new ResultImage. To actually initialize the returned ResultImage, use ResultImage

Parameters:

`imW` - the width of the image

`imH` - the height of the image

Method Detail

computeResults

```
public void computeResults(Img img)
```

This method goes through an `Img` and finds which pixels appear to match the color of the vision target.

Parameters:

`img` - the image to process and find matching Target-colored pixels

isTarget

```
public boolean isTarget(int x,
                        int y)
```

Parameters:

`x` - The X coordinate, in pixels

`y` - The Y coordinate, in pixels

Returns:

com._604robotics.robot2012.vision

Class Result

java.lang.Object
    com._604robotics.robot2012.vision.Result

Direct Known Subclasses:

Result.AntiResult, Result.PlusResult

```
public abstract class Result
extends java.lang.Object
```

This class stores one tile of "is in target" data. If there are no matches for the target, a `Result.AntiResult` is used. If there are matching pixels, a `Result.PlusResult` is used.

Nested Class Summary

Nested Classes

Modifier and Type	Class and Description
static class	**Result.AntiResult** A result indicating that it is unlikely that the target lies in the indicated tile
static class	**Result.PlusResult** A result indicating that it is likely that the target lies in the indicated tile

Constructor Summary

Constructors

Constructor and Description
Result()

Method Summary

Methods

Modifier and Type	Method and Description
boolean	**hasPlus**()
boolean	**plusAt**(int x, int y)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Result

```
public Result()
```

Method Detail

hasPlus

```
public boolean hasPlus()
```

Returns:

whether there are any pixels matching the color of the target or not

plusAt

```
public boolean plusAt(int x,
                      int y)
```

Parameters:

`x` - the X coordinate (within the tile, not the image)

`y` - the Y coordinate (within the tile, not the image)

Returns:

whether or not the pixel at the given location matches the Target color

com._604robotics.robot2012.vision

Class Result.AntiResult

java.lang.Object
  com._604robotics.robot2012.vision.Result
    com._604robotics.robot2012.vision.Result.AntiResult

Enclosing class:

Result

```
public static class Result.AntiResult
extends Result
```

A result indicating that it is unlikely that the target lies in the indicated tile

Nested Class Summary

Nested classes/interfaces inherited from class com._604robotics.robot2012.vision.Result

Result.AntiResult, Result.PlusResult

Constructor Summary

Constructors

Constructor and Description
Result.AntiResult()

Method Summary

Methods inherited from class com._604robotics.robot2012.vision.Result

hasPlus, plusAt

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Result.AntiResult

```
public Result.AntiResult()
```

com._604robotics.robot2012.vision

Class LinearRegression

java.lang.Object
    com._604robotics.robot2012.vision.LinearRegression

```
public class LinearRegression
extends java.lang.Object
```

Accepts a sequence of pairs of real numbers and computes the best fit (least squares) line y = ax + b through the set of points. Also computes the correlation coefficient and the standard error of the regression coefficients.

Nested Class Summary

Nested Classes

Modifier and Type	Class and Description
static class	**LinearRegression.BackwardsRegressionResult** A regression result that, instead of having y as a function of x has x as a function of y.
static class	**LinearRegression.RegressionResult** A regression result that indicates the line that best matches a given set of data.

Constructor Summary

Constructors

Constructor and Description
LinearRegression()

Method Summary

Methods

Modifier and Type	Method and Description
static **LinearRegression.BackwardsRegressionResult**	**getBackwardsRegression**(double[] y, double[] x) This returns a regression result that, instead of having y as a function of x has x as a function of y.
static **LinearRegression.RegressionResult**	**getRegression**(double[] x, double[] y) This function computes the linear regression of a set of x and y values.
static P **oin2d**	**solve**(**LinearRegression.RegressionResult** a, **LinearRegression.RegressionResult** b) Computes the intersection of two RegressionResults

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

LinearRegression

```
public LinearRegression()
```

Method Detail

getBackwardsRegression

```
public static LinearRegression.BackwardRegressionResult getBackwardRegression(double[] y,
                                                                               double[] x)
```

This returns a regression result that instead of having y as a function of x has x as a function of y.

Parameters:

`y` - the list of Y values

`x` - the list of X values

Returns:

getRegression

```
public static LinearRegression.RegressionResult getRegression(double[] x,
                                                               double[] y)
```

This function computes the linear regression of a set of x and y values.

It is largely taken from: http://introcs.cs.princeton.edu/java/97data/LinearRegression.java.html

Parameters:

`x` - An array of X values

`y` - An array of Y values

Returns:

solve

```
public static Point2d solve(LinearRegression.RegressionResult a,
                            LinearRegression.RegressionResult b)
```

Computes the intersection of two RegressionResults

Parameters:

`a` - A RegressionResult

`b` - A RegressionResult

Returns:

The intersection

com._604robotics.robot2012.vision

Class Img

java.lang.Object
    com._604robotics.robot2012.vision.Img

```
public class Img
extends java.lang.Object
```

A simple class for accessing 2d data in a 1d array, with bounds checking.

Constructor Summary

Constructors

Constructor and Description
`Img(int[] dat, int w, int h)` A constructor to make an Img
`Img(int w, int h)` A constructor to make an Img
`Img(java.awt.image.Raster raster)` A constructor to make an Img
`Img(java.awt.image.Raster raster, int[] buff)` A constructor to make an Img

Method Summary

Methods

Modifier and Type	Method and Description
`int`	`get(int x, int y)`
`boolean`	`set(int x, int y, int k)`

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

Img

```
public Img(int[] dat,
   int w,
   int h)
```

A constructor to make an Img

Parameters:

`dat` - data array

`w` - width

`h` - height

Img

```
public Img(java.awt.image.Raster raster,
   int[] buff)
```

A constructor to make an Img

Parameters:

`raster` - a raster storing original image data

`buff` - an array to store the image data into

Img

```
public Img(java.awt.image.Raster raster)
```

A constructor to make an Img

Parameters:

`raster` - a raster storing original image data

Img

```
public Img(int w,
   int h)
```

A constructor to make an Img

Parameters:

`w` -

`h` -

Method Detail

get

```
public int get(int x,
      int y)
```

Parameters:

`x` - the X coordinate

`y` - the Y coordinate

Returns:

an integer holding an RGB value

set

```
public boolean set(int x,
          int y,
          int k)
```

Parameters:

`x` - the X coordinate

`y` - the Y coordinate

`k` - an integer holding an RGB value

Returns:

a boolean if the value was set or not

com._604robotics.robot2012.vision

Class Quad

java.lang.Object
    com._604robotics.robot2012.vision.Quad

```
public class Quad
extends java.lang.Object
```

A class representing a Quadrilateral, with four corner points.

Constructor Summary

Constructors

Constructor and Description
Quad(**Point2d** topLeft, **Point2d** topRight, **Point2d** bottomLeft, **Point2d** bottomRight)

Method Summary

Methods

Modifier and Type	Method and Description
double	**getAvgHeight**()
double	**getAvgWidth**()
double	**getAvgX**()
double	**getAvgY**()
double	**getMaxX**()
double	**getMaxY**()
double	**getMinX**()
double	**getMinY**()
java.lang.String	**toString**()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

Quad

```
public Quad(Point2d topLeft,
    Point2d topRight,
    Point2d bottomLeft,
    Point2d bottomRight)
```

Parameters:

topLeft -

topRight -

bottomLeft -

bottomRight -

Method Detail

toString

```
public java.lang.String toString()
```

Overrides:

`toString` in class `java.lang.Object`

getAvgWidth

```
public double getAvgWidth()
```

Returns:

the average width of this Quad

getAvgHeight

```
public double getAvgHeight()
```

Returns:

the average height of this Quad

getAvgX

```
public double getAvgX()
```

Returns:

the average X values of this Quad

getAvgY

```
public double getAvgY()
```

Returns:

the average Y values of this Quad

getMinX

```
public double getMinX()
```

Returns:

the minimum X value of this Quad

getMaxX

```
public double getMaxX()
```

Returns:

the maximum X value of this Quad

getMinY

```
public double getMinY()
```

Returns:

the minimum Y value of this Quad

getMaxY

```
public double getMaxY()
```

Returns:

the maximum Y value of this Quad

com._604robotics.robot2012.vision

Class DistanceCalculations

java.lang.Object
    com._604robotics.robot2012.vision.DistanceCalculations

```
public class DistanceCalculations
extends java.lang.Object
```

This code does the 2D-to-3D calculations

Field Summary

Fields

Modifier and Type	Field and Description
static double	**cameraPixelHeight** The size of the Axis camera, in pixels
static double	**cameraPixelWidth** The size of the Axis camera, in pixels

Constructor Summary

Constructors

Constructor and Description
DistanceCalculations()

Method Summary

Methods

Modifier and Type	Method and Description
double	**getAngleOfTarget**(**Quad** q, double z) This function gets the direction the target is facing, relative to the camera.
Target	**getApproximationOfTarget**(**Quad** quad) A method that tries to find the most likely location for the vision target to lie in 3D space
Point3d	**getRelXYZOfTarget**(**Quad** q) Remember that this requires the camera to be "perfectly" flat, and the targets to be "perfectly" vertical.

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

cameraPixelHeight

```
public static final double cameraPixelHeight
```

The size of the Axis camera, in pixels

See Also:

Constant Field Values

cameraPixelWidth

```
public static final double cameraPixelWidth
```

The size of the Axis camera, in pixels

See Also:

Constant Field Values

Constructor Detail

DistanceCalculations

```
public DistanceCalculations()
```

Method Detail

getAngleOfTarget

```
public double getAngleOfTarget(Quad q,
                               double z)
```

This function gets the direction the target is facing, relative to the camera. It is imperfect, and half-assumes a simple orthographic projection (which is not quite like real life). If it causes issues (which the accuracy of this function doesn't need to be very high), we can fix it later.

Returns:

the resulting angle in radians.

getApproximationOfTarget

```
public Target getApproximationOfTarget(Quad quad)
```

A method that tries to find the most likely location for the vision target to lie in 3D space

Parameters:

`quad` - a quadrilateral with corners indicating the corners of the target

Returns:

a Target as an estimation of

getRelXYZOfTarget

```
public Point3d getRelXYZOfTarget(Quad q)
```

Remember that this requires the camera to be "perfectly" flat, and the targets to be "perfectly" vertical. A new function will probably need to be created for use on the robot. That, or we'll need to manipulate the points based on camera angle.

Returns:

a Point3d holding the X, Y, and Z of the target, relative to the camera.

com._604robotics.robot2012.vision

Class VisionDisp

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.JPanel
          com._604robotics.robot2012.vision.VisionDisp

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible

---

```
public class VisionDisp
extends javax.swing.JPanel
```

This class is used to display a camera image and some debug information along with it.

See Also:

Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class javax.swing.JPanel

javax.swing.JPanel.AccessibleJPanel

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

Nested classes/interfaces inherited from class java.awt.Container

java.awt.Container.AccessibleAWTContainer

Nested classes/interfaces inherited from class java.awt.Component

java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy

Field Summary

Fields

Modifier and Type	Field and Description
java.awt.image.BufferedImage	**image** The background image, as received from the camera

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

Constructors

Constructor and Description
VisionDisp( ) A default constructor that sets this up as a 640x480 display

Method Summary

Methods

Modifier and Type	Method and Description
void	**paint**(java.awt.Graphics g) Paints this VisionDisp.

Methods inherited from class javax.swing.JPanel

getAccessibleContext, getUI, getUIClassID, paramString, setUI, updateUI

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paintBorder, paintChildren, paintComponent, paintImmediately, paintImmediately, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setLayout, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

image

```
public java.awt.image.BufferedImage image
```

The background image, as received from the camera

Constructor Detail

VisionDisp

```
public VisionDisp()
```

A default constructor that sets this up as a 640x480 display

Method Detail

paint

```
public void paint(java.awt.Graphics g)
```

Paints this `VisionDisp`.

If available, this draws the camera image, resulting tiled red-and-green "isTarget" image, target corners, and target sides

Overrides:

`paint` in class `javax.swing.JComponent`

See Also:

`JComponent.paint(java.awt.Graphics)`

com._604robotics.robot2012.vision

Class AABB

java.lang.Object
    com._604robotics.robot2012.vision.AABB

```
public class AABB
extends java.lang.Object
```

An Axis-Aligned Bounding Box. This stores two opposite corner values of a rectangle that has perfectly vertical and horizontal sides.

Field Summary

Fields

Modifier and Type	Field and Description
int	x1
int	x2
int	y1
int	y2

Constructor Summary

Constructors

Constructor and Description
AABB(int x1, int y1, int x2, int y2)

Method Summary

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

x1

```
public int x1
```

y1

```
public int y1
```

x2

```
public int x2
```

y2

```
public int y2
```

Constructor Detail

AABB

```
public AABB(int x1,
    int y1,
    int x2,
    int y2)
```

Parameters:

`x1` - - lowest x value on the rectangle

`y1` - - lowest y value on the rectangle

`x2` - - highest x value on the rectangle

`y2` - - highest y value on the rectangle

com._604robotics.tcpcommunicator

Class TcpCommunicator

java.lang.Object
    com._604robotics.tcpcommunicator.TcpCommunicator

All Implemented Interfaces:

java.lang.Runnable

```
public class TcpCommunicator
extends java.lang.Object
implements java.lang.Runnable
```

Server class for the vision data transfer protocol.

Constructor Summary

Constructors

Constructor and Description
TcpCommunicator() Initializes a new TcpCommunicator.
TcpCommunicator(java.lang.String ip) Initializes a new TcpCommunicator with the specified robot IP address.
TcpCommunicator(java.lang.String ip, int port) Initializes a new TcpCommunicator with the specified robot IP address and port.
TcpCommunicator(java.lang.String ip, int port, boolean debug) Initializes a new TcpCommunicator with the specified robot IP address, port, and debug mode.

Method Summary

Methods

Modifier and Type	Method and Description
void	**down**() Disables the TcpCommunicator.
void	**forceQuit**() Interrupts the TcpCommunicator thread, forcing it to quit.
boolean	**isEnabled**() Checks whether or not the TcpCommunicator has been enabled.
boolean	**isRunning**() Checks whether or not the TcpCommunicator thread is currently running.
static void	**main**(java.lang.String[] args) For testing purposes.
void	**run**() Don't use this to launch the server; use up() instead.
void	**up**() Enables the TcpCommunicator, launching the thread.
void	**writePoints**(**Target**[] points) Writes the specified points to the stream.

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

TcpCommunicator

```
public TcpCommunicator()
```

Initializes a new TcpCommunicator. By default, the robot IP address is set to "10.6.4.2", the port is set to 3333, and the debug mode is set to TRUE.

TcpCommunicator

```
public TcpCommunicator(java.lang.String ip)
```

Initializes a new TcpCommunicator with the specified robot IP address. By default, the port is set to 3333 and the debug mode is set to TRUE.

Parameters:

`ip` - The IP address of the robot.

TcpCommunicator

```
public TcpCommunicator(java.lang.String ip,
               int port)
```

Initializes a new TcpCommunicator with the specified robot IP address and port. By default, the debug mode is set to TRUE.

Parameters:

`ip` - The IP address of the robot.

`port` - The port to connect to.

TcpCommunicator

```
public TcpCommunicator(java.lang.String ip,
               int port,
               boolean debug)
```

Initializes a new TcpCommunicator with the specified robot IP address, port, and debug mode.

Parameters:

`ip` - The IP address of the robot.

`port` - The port to connect to.

`debug` - Print debug info?

Method Detail

isEnabled

```
public boolean isEnabled()
```

Checks whether or not the TcpCommunicator has been enabled.

Returns:

Whether or not the TcpCommunicator has been enabled.

isRunning

```
public boolean isRunning()
```

Checks whether or not the TcpCommunicator thread is currently running.

Returns:

Whether or not the TcpCommunicator thread is currently running.

up

```
public void up()
```

Enables the TcpCommunicator, launching the thread.

down

```
public void down()
```

Disables the TcpCommunicator.

forceQuit

```
public void forceQuit()
```

Interrupts the TcpCommunicator thread, forcing it to quit. Use only in emergencies!

writePoints

```
public void writePoints(Target[] points)
```

Writes the specified points to the stream. If there is no robot currently connected, it fails silently and discards the points into the ether.

Parameters:

`points` - An array of Targets to write.

run

```
public void run()
```

Don't use this to launch the server; use up() instead. This implements the run() method of type Runnable, allowing this to be run as a thread. For internal use!

Specified by:

`run` in interface `java.lang.Runnable`

main

```
public static void main(java.lang.String[] args)
```

For testing purposes. Run this as an application, and it will connect to 127.0.0.1 and stream arbitrary data for testing purposes.

Parameters:

`args` - Command-line arguments. Not currently used.