概述

2014-2-14

要定义拖动滑块对象，首先调用create函数，参数为图片路径，分别是滑块滑道图片路径、滑块滑动后覆盖滑道图片路径和滑块图片路径；之后设置锚点，并设置范围最小值和设置范围最大值，设置位置后给拖动注册拖动事件接受函数。在拖动事件中，可以通过pSlider->getValue())->getCString()来获取目前所在位置的值，运行效果如图3-19所示。

bool CCControlSliderTest::init()

{

if (CCControlScene::init())

{

CCSize screenSize = CCDirector::sharedDirector()->getWinSize();

// Add a label in which the slider value will be displayed

/// 定¡§义°?标À¨º签?

m\_pDisplayValueLabel = CCLabelTTF::create("Move the slider thumb!\nThe lower slider is restricted." ,"Marker Felt", 32);

m\_pDisplayValueLabel->retain();

m\_pDisplayValueLabel->setAnchorPoint(ccp(0.5f, -1.0f));

m\_pDisplayValueLabel->setPosition(ccp(screenSize.width / 1.7f, screenSize.height / 2.0f));

addChild(m\_pDisplayValueLabel);

// Add the slider

/// 定¡§义°?一°?个?拖ª?动¡¥滑?块¨¦控?件t

CCControlSlider \*slider = CCControlSlider::create("extensions/sliderTrack.png","extensions/sliderProgress.png" ,"extensions/sliderThumb.png");

slider->setAnchorPoint(ccp(0.5f, 1.0f));

slider->setMinimumValue(0.0f); // Sets the min value of range

slider->setMaximumValue(5.0f); // Sets the max value of range

slider->setPosition(ccp(screenSize.width / 2.0f, screenSize.height / 2.0f + 16));

slider->setTag(1);

// When the value of the slider will change, the given selector will be call

/// 当Ì¡À拖ª?动¡¥控?件t时º¡À，ê?调Ì¡Â用®?此ä?函¡¥数ºy

slider->addTargetWithActionForControlEvents(this, cccontrol\_selector(CCControlSliderTest::valueChanged), CCControlEventValueChanged);

CCControlSlider \*restrictSlider = CCControlSlider::create("extensions/sliderTrack.png","extensions/sliderProgress.png" ,"extensions/sliderThumb.png");

restrictSlider->setAnchorPoint(ccp(0.5f, 1.0f));

restrictSlider->setMinimumValue(0.0f); // Sets the min value of range

restrictSlider->setMaximumValue(5.0f); // Sets the max value of range

restrictSlider->setMaximumAllowedValue(4.0f);

restrictSlider->setMinimumAllowedValue(1.5f);

restrictSlider->setValue(3.0f);

restrictSlider->setPosition(ccp(screenSize.width / 2.0f, screenSize.height / 2.0f - 24));

restrictSlider->setTag(2);

//same with restricted

restrictSlider->addTargetWithActionForControlEvents(this, cccontrol\_selector(CCControlSliderTest::valueChanged), CCControlEventValueChanged);

addChild(slider);

addChild(restrictSlider);

return true;

}

return false;

}

void CCControlSliderTest::valueChanged(CCObject \*sender, CCControlEvent controlEvent)

{

CCControlSlider\* pSlider = (CCControlSlider\*)sender;

// Change value of label.

if(pSlider->getTag() == 1)

m\_pDisplayValueLabel->setString(CCString::createWithFormat("Upper slider value = %.02f", pSlider->getValue())->getCString());

if(pSlider->getTag() == 2)

m\_pDisplayValueLabel->setString(CCString::createWithFormat("Lower slider value = %.02f", pSlider->getValue())->getCString());

}