

Name of Experiment: Camera Operation(Capture Event)

Exp No:WP6

Background: Student should have basic knowledge about C#.

Summary: This experiment is completely independent from previous experiment WP5, and it will help you to understand the camera operation and development of dedicated camera application.

Learning Objective: Student will learn the basic steps in order to access the camera hardware for the device.

Target Platforms: This application is tested on Windows Phone Emulator and Windows Phone (Lumia 800).

Procedure:

Step1. Repeat all steps [1-6] as in experiment no WP1.

Step2. Now, change the UI design from Portrait to Landscape, by setting the **SupportedOrientations="Landscape" Orientation="LandscapeLeft"** in the xaml code. Also, set the **shell:SystemTray.IsVisible="False"** to hide the SystemTray.

Step3. Set the UI accordingly in order to capture the image in full screen.[Refer fig no 1 in Snapshots]

Step4. Add Reference of Microsoft.Xna.Framework in the current project directory in order to use the MediaLibrary class and its delegate methods.[Refer fig no. 2-3]

Step5. Now, define some global object like Photocamera object, savedPhoto counter, and a MediaLibrary object before MainPage constructor.

Step6. Override two methods named OnNavigatedTo(System.Windows.Navigation.NavigationEventArgs e) and OnNavigatedFrom(System.Windows.Navigation.NavigationEventArgs e).[Refer the Source code section]

Step7. Initialize the PhotoCamera object and set the source to VideoBrush.

Step8. Now, add some Photocamera event handlers like (PhotoCamera)object.Initialize Event, (PhotoCamera)object.CaptureCompleted Event and (PhotoCamera)object.CaptureImageAvailable Event.[Refer source code MainPage.xaml.cs]

Step9. Also, add button Click handler in order to capture the image from camera device.

Step10. Now save all changes made to the experiment.

Step11. Press F5,to start the Windows Emulator for debugging purpose.[Refer fig no 4-5]

Source Code	Comments
<p>MainPage.xaml</p> <pre> <!--LayoutRoot is the root grid where all page content is placed--> <Grid x:Name="LayoutRoot" Background="Transparent"> <Grid.ColumnDefinitions> <ColumnDefinition Width="640"/> <ColumnDefinition Width="160"/> </Grid.ColumnDefinitions> <!--TitlePanel contains the name of the application and page title--> <Canvas x:Name="viewfinderCanvas" Width="640" Height="480" HorizontalAlignment="Left" > <!--Camera viewfinder --> <Canvas.Background> <!--VideoBrush for Image drawing--> <VideoBrush x:Name="viewfinderBrush" /> </Canvas.Background> </Canvas> <!--Button StackPanel to the right of viewfinder--> <StackPanel Grid.Column="1" > <!--Name= "ShutterButton" Content= "SH"--> <Button x:Name="ShutterButton" Content="SH" Click="ShutterButton_Click" FontSize="26" FontWeight="ExtraBold" Height="75" /> </StackPanel> <!--Used for debugging --> <TextBlock Height="40" HorizontalAlignment="Left" Margin="8,428,0,0" Name="txtDebug" VerticalAlignment="Top" Width="626" FontSize="24" FontWeight="ExtraBold" /> </Grid> </pre> <p>MainPage.xaml.cs</p> <pre> using System.Windows; using Microsoft.Phone.Controls; using Microsoft.Devices; using System; using Microsoft.Xna.Framework.Media; namespace test_cap { public partial class MainPage : PhoneApplicationPage { PhotoCamera cam = null; int savedPhoto=0; MediaLibrary library = new MediaLibrary(); // Constructor public MainPage() { InitializeComponent(); } } </pre>	<p>← VideoBrush for Image drawing</p> <p>←Name= "ShutterButton" Content= "SH"</p> <p>←Microsoft.Xna.Framework.Media</p>

```

protected override void
OnNavigatedTo(System.Windows.Navigation.NavigationEventArgs e)
{
    //base.OnNavigatedTo(e);
    if (PhotoCamera.IsCameraTypeSupported(CameraType.Primary) ==
true)
    {
        cam = new Microsoft.Devices.PhotoCamera(CameraType.Primary);
        viewfinderBrush.SetSource(cam);
        cam.Initialized += new
System.EventHandler<CameraOperationCompletedEventArgs>(cam_Initialized
);
        cam.CaptureImageAvailable += new
System.EventHandler<ContentReadyEventArgs>(cam_CaptureImageAvailable)
;
        cam.CaptureCompleted+=new
System.EventHandler<CameraOperationCompletedEventArgs>(cam_CaptureC
ompleted);
    }
    else {
        Deployment.Current.Dispatcher.BeginInvoke(delegate()
        {
            this.txtDebug.Text = "Camera is not available ";
        });
        ShutterButton.IsEnabled = false;
    }
}
protected override void
OnNavigatedFrom(System.Windows.Navigation.NavigationEventArgs e)
{
    //base.OnNavigatedFrom(e);
    if (cam != null)
    { cam.Dispose();
    cam.Initialized -= cam_Initialized;
    cam.CaptureImageAvailable -= cam_CaptureImageAvailable;
    cam.CaptureCompleted -= cam_CaptureCompleted;

    }

}
public void cam_Initialized(object sender ,
CameraOperationCompletedEventArgs e) {
    if (e.Succeeded)
    {
        this.Dispatcher.BeginInvoke(delegate()
        {
            txtDebug.Text = "camera is initialized";
        });
    }
}
public void cam_CaptureCompleted(object sender,
CameraOperationCompletedEventArgs e) {
    savedPhoto++;
}

```

← Microsoft.Devices

```

    }
    public void cam_CaptureImageAvailable(object sender,
ContentReadyEventArgs e) {
        string filename = savedPhoto + ".jpg";
library.SavePictureToCameraRoll(filename, e.ImageStream);

    Deployment.Current.Dispatcher.BeginInvoke(delegate()
        {
this.txtDebug.Text = "Picture has been saved for you";
        });
    }

private void ShutterButton_Click(object sender, RoutedEventArgs e)
    {
        try{
            if (cam != null) {
                cam.CaptureImage(); }
        }
        catch(Exception ex){
            this.Dispatcher.BeginInvoke(delegate()
                {
txtDebug.Text = ex.Message;
                });
        }
    }
}
}

```

←System

Screenshots

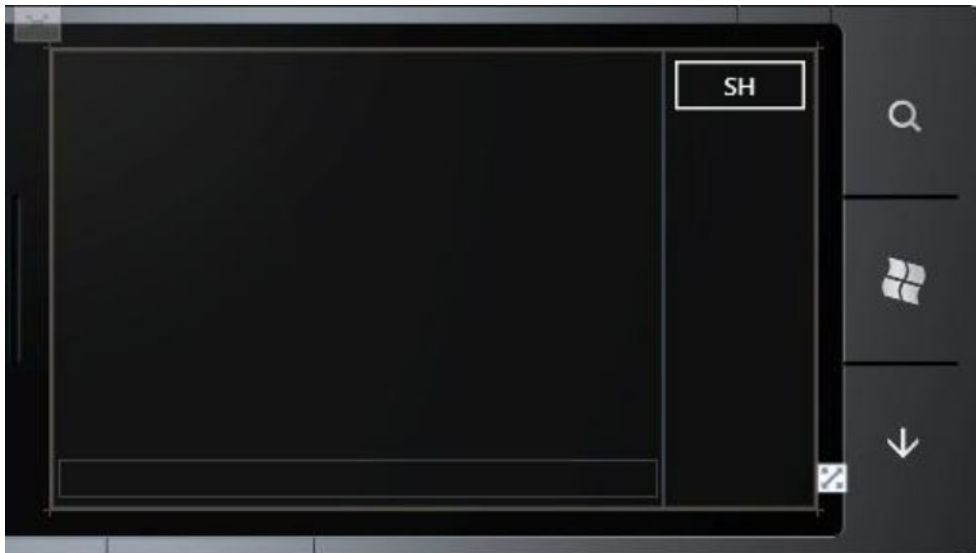


Fig no 1 UI Design for Camera Operation

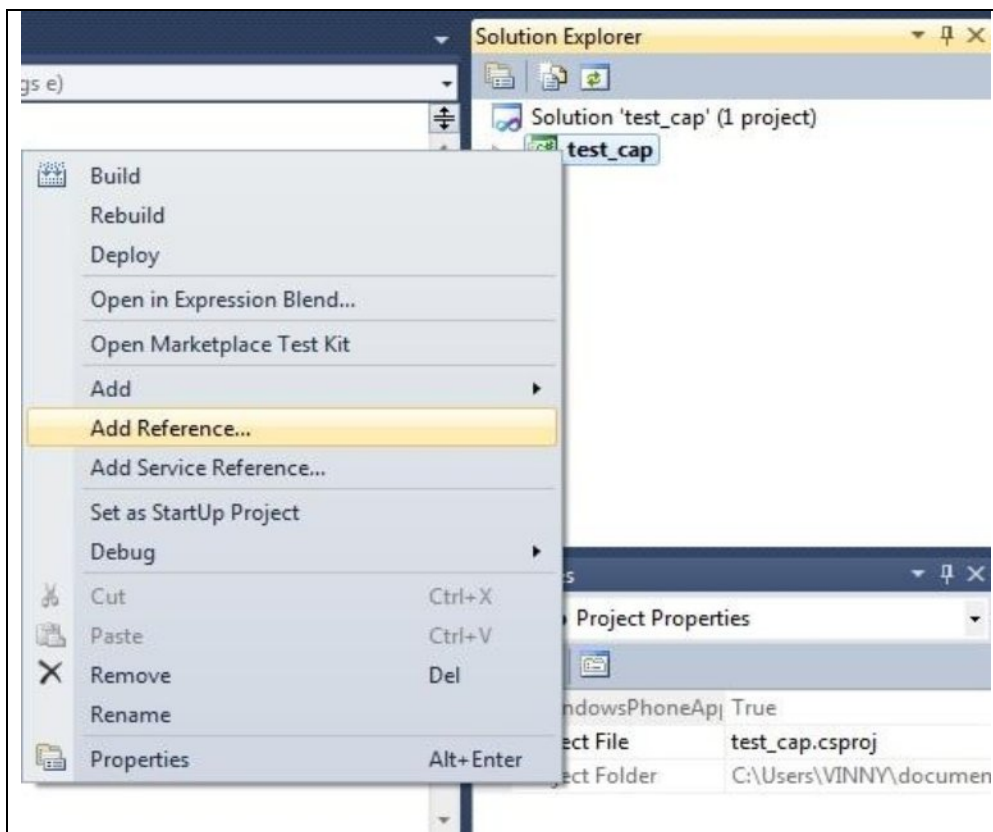


Fig no 2 Add Reference

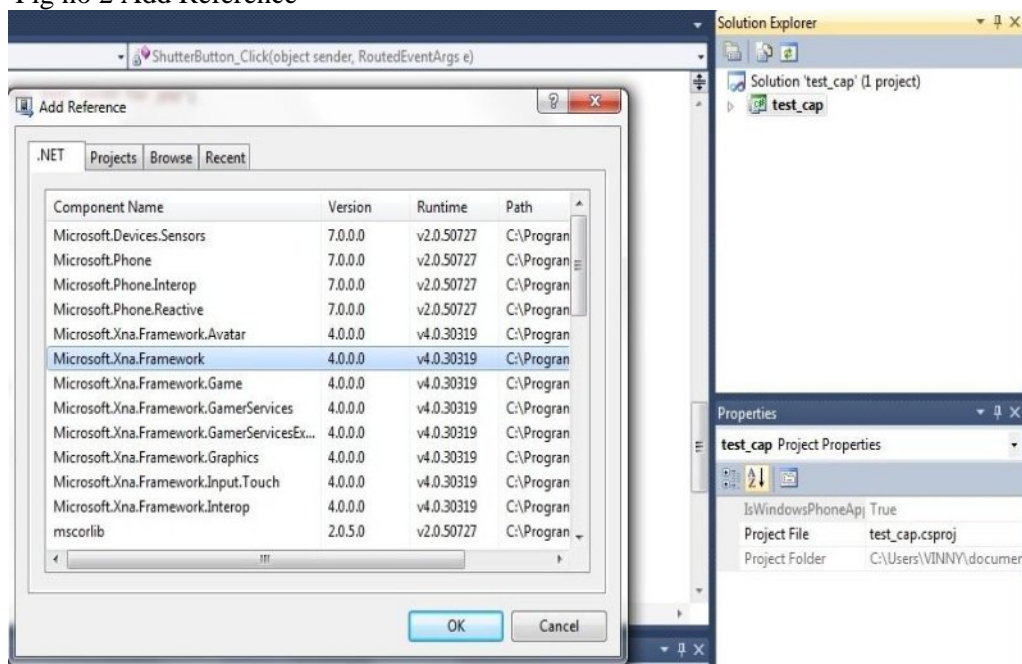


Fig no 3 Adding Microsoft.Xna.Framework



Fig no 4 Experiment running on Windows Emulator

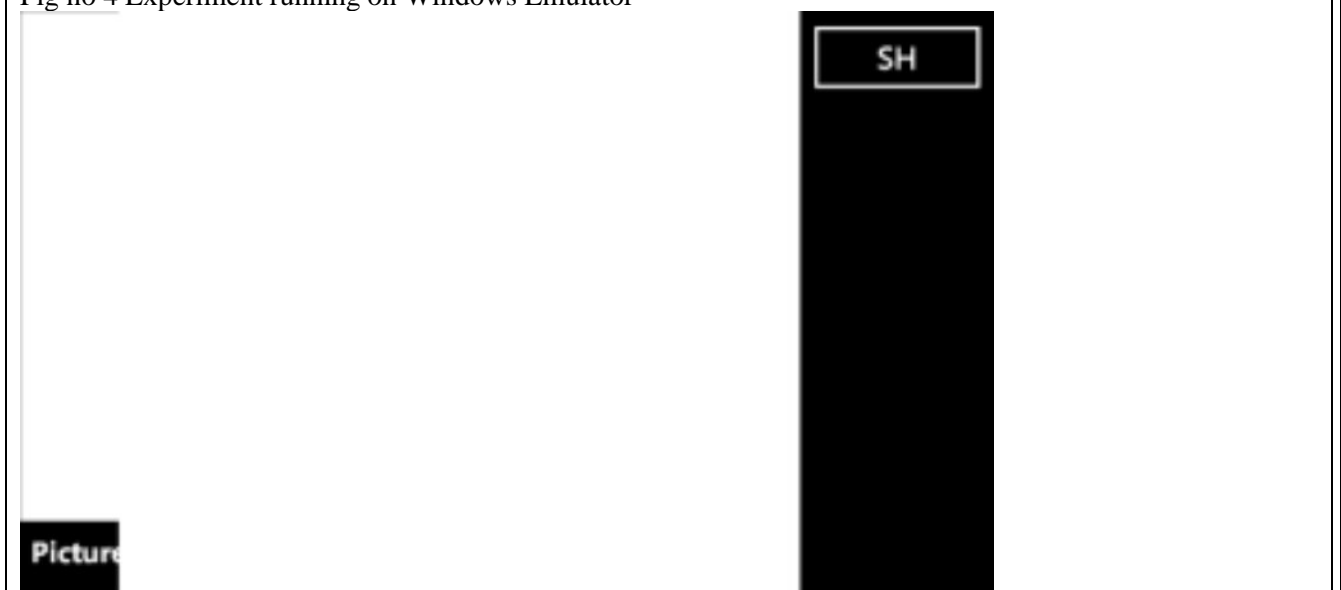


Fig no 5 After capturing the Image from camera

Observations: It is observed that for accessing the basic functionality of the camera device these above steps must be performed.