

FingerEyes-Xr for HTML5 Tutorials - 01

TMS 활용하기

2015 년 3 월 2 일, 1 차 배포



TMS 활용하기

FingerEyes-Xr은 OSGeo 스펙 중 하나인 TMS를 지원합니다. TMS는 Tile Map Service로써 일반적으로 지도에 대한 격자 형태의 이미지를 미리 만들어 놓고 Pyramid와 Row 그리고 Column에 대한 인덱스로 접근할 수 있는 URL을 통해 지도를 사용자에게 빠르게 제공하는 것을 목적으로 합니다.

이 글은 VWorld에서 TMS 방식으로 서비스 되는 배경지도를 FingerEyes-Xr에서 사용하는 내용을 담고 있습니다. FingerEyes-Xr은 Flex 버전과 HTML5 버전이 존재하며 이 글은 HTML5에 대해 글입니다. FingerEyes-Xr for HTML5에 대한 소스 코드는 GitHub에서 다운로드 받을 수 있으며 URL은 <https://github.com/FingerEyes-Xr/src> 입니다. 아래의 화면은 FingerEyes-Xr for HTML5에 대한 GitHub 웹 화면입니다.

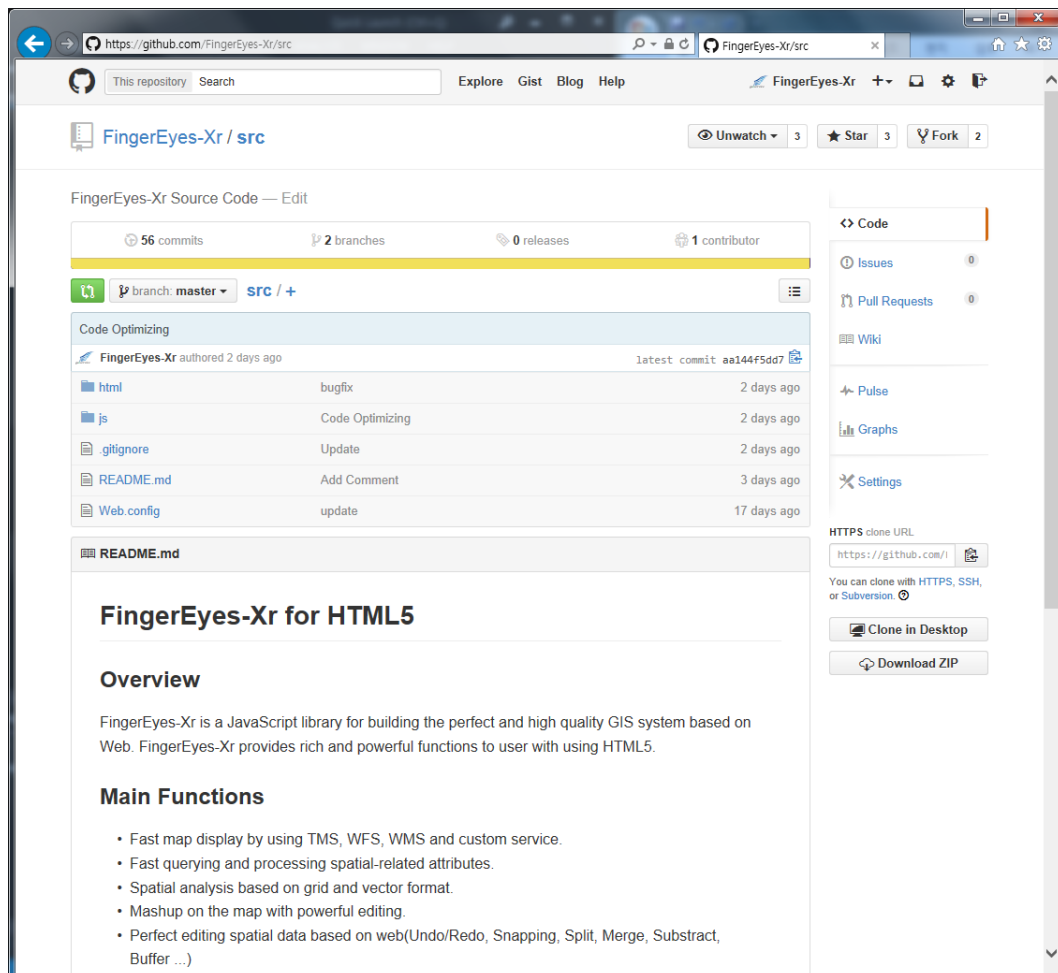


그림 1. FingerEyes-Xr for HTML5에 대한 GitHub

필자는 이 글에 대한 예제 코드를 VisualStudio 2012를 이용하여 작성 하였습니다. 독자 여러분들이 어떤 툴을 사용하든 문제는 없겠으나 VisualStudio 2012에서 제공하는 JavaScript 디버깅과 기본적으로 제공하는 웹서버(IIS)를 통해 웹 기반의 프로그래밍을 편리하고 효율적으로 진행할 수 있었기에 선택하였습니다. 참고로 웹 기반의 프로그래밍은 웹서버를 이용하여 URL 형태로 접근하지 않으면 올바르게 실행되지 않

으므로 반드시 웹서버를 통해 URL 형태로 접근하시기 바랍니다.

이제 본적으로 TMS 방식으로 서비스 되는 배경지도 중의 하나인 VWorld를 FingerEyes-Xr에서 사용해 보는 코드를 작성해 보겠습니다. 먼저 map.html 파일을 웹서버에서 접근할 수 있는 곳에 생성합니다. 즉, URL을 통해 웹브라우저에서 실행할 수 있는 경로에 생성합니다. 그리고 다음처럼 입력합니다.

```
01 <html xmlns="http://www.w3.org/1999/xhtml">
02 <head>
03   <title>FingerEyes-Xr : VWorld Basemap</title>
04 </head>
05 <body>
06
07 </body>
08 </html>
```

코드 1. 초기 map.html

위의 코드에서 4번 라인 밑에 다음 코드를 추가합니다.

```
01 <script src="http://www.geoservice.co.kr/z/Xr.min.1.0.js"></script>
02
03 <script type="text/javascript">
04
05 </script>
```

코드 2. 기본 Script

1번 코드에서 CDN(Content Delivery Network)을 사용하여 FingerEyes-Xr JavaScript 라이브러리에 대한 소스코드를 추가 하고 있습니다. 그리고 4번 코드에 새롭게 추가될 JavaScript 코드가 입력될 것입니다.

UI를 구성하기 위해 <body> 부분을 다음처럼 변경합니다.

```
01 <body>
02   <div id="mainLayout">
03     <div id="mapDiv"></div>
04
05     <div id="title">
06       FingerEyes-Xr for HTML5 :
07       <font color="#349bd6">Pan/Zoom/Rotate Example</font>
08       <br />
09       <input type="button" value="확대" onclick="onZoomInMap();">
10       <input type="button" value="축소" onclick="onZoomOutMap();">
11       <input type="button" value="10도 회전" onclick="onRotateMap();">
12       <input type="button" value="-10도 회전" onclick="onRevRotateMap();">
13       <font size="2" color="#ffff00">
14       <span> 배경지도 : 브이월드 (VWorld)</span></font>
15       <input type="checkbox" onchange="onBaseMapChanged(this)" />
16       <font size="3" color="#ffffff"><span>항공영상</span></font>
17     </div>
18   </div>
19 </body>
```

코드 3. 기본 UI 구성

특히 Id가 mapDiv인 DIV 요소에 맵(Map)이 표시될 것입니다. 일단 여기까지 작성하고 실행해 보면 다음과 같은 화면이 나타나는 것을 볼 수 있습니다.

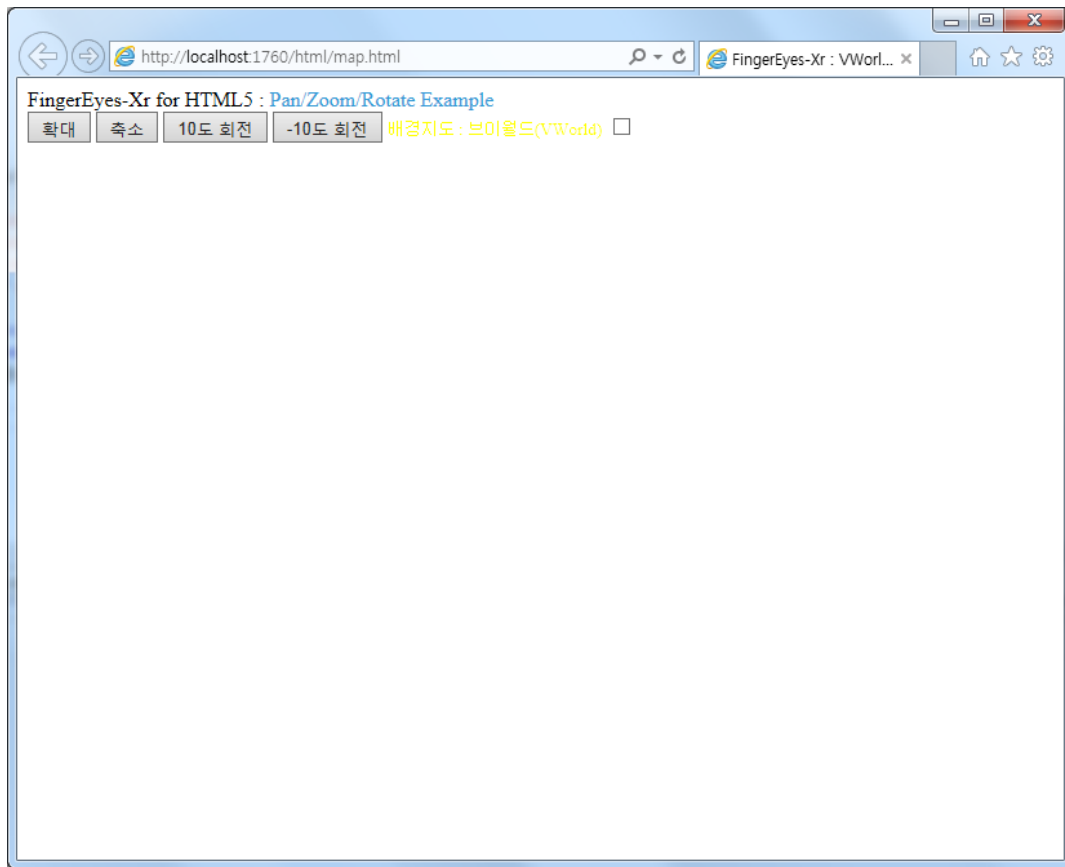


그림 2. 기본 UI 구성에 대한 실행 화면

Button이 4개 있고 하나의 CheckBox가 있습니다. 각각의 버튼은 지도를 코드를 통해 확대, 축소, 회전하는 것이고 CheckBox는 체크가 바뀔 때마다 지도를 다른 형태로 전환되는데 사용됩니다.

이제 만든 페이지에 스타일을 적용해 보도록 하겠습니다. <head> 바로 밑에 아래의 스타일 코드를 추가합니다.

```

01 <style>
02 body
03 {
04     margin:0px;
05     padding:0px;
06 }
07
08 #mainLayout
09 {
10     position:relative;
11     width:100%;
12     height:100%;
13     border: none;
14 }
15
    
```

```

16 #mapDiv {
17     top:0px;
18     left:0px;
19     position:relative;
20     width:100%;
21     height:100%;
22     border: none;
23     overflow:auto;
24 }
25
26 #title {
27     top:12px;
28     left:12px;
29     padding: 12px;
30     position:absolute;
31     background:rgba(0,0,0,0.7);
32     border:none;
33     overflow:auto;
34     border-radius: 12px;
35     font-size: 24px;
36     color: #ffffff;
37     font-family: "Arial";
38 }
</style>

```

코드 4. UI에 대한 스타일 적용

스타일이 적용된 페이지는 다음과 같습니다.

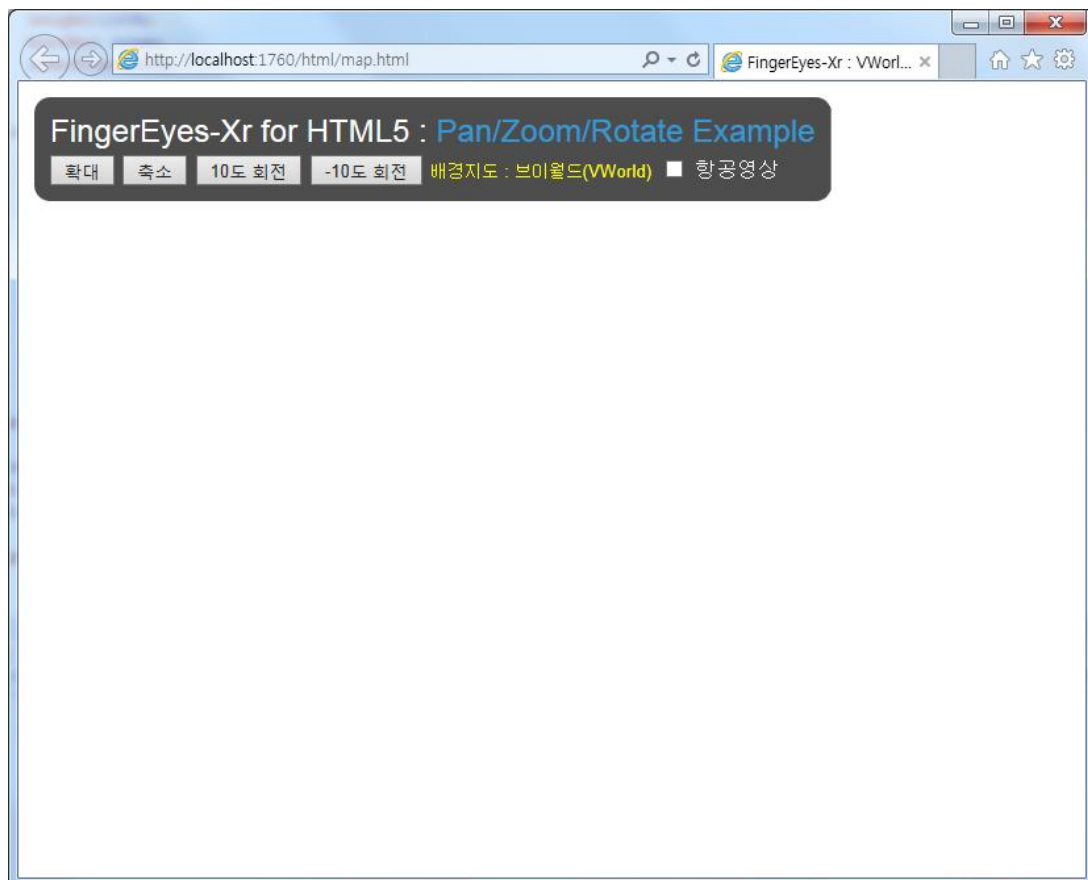


그림 3. 스타일이 적용된 UI

이제 UI와 스타일 적용이 완료되었으므로, 이제 코드를 작성해 보겠습니다. **코드 2. 기본 Script**에서 4번 줄에 다음 코드를 추가합니다.

```
01 var map = null;
```

코드 5. DIV와 연결될 지도 객체 정의

이 map 변수는 지도 객체에 대한 참조로 사용됩니다. 이제 <body>에 대한 onload 이벤트를 load() 함수로 지정합니다.

```
01 <body onload="load()" ">
```

코드 6. Body 요소의 onload 이벤트 지정

그리고 load 함수를 다음처럼 추가합니다.

```
01 function load() {
02     map = new Xr.Map("mapDiv", {});
03
04
05
06 }
```

코드 7. Body의 onload 이벤트 함수

2번 코드는 id가 mapDiv인 DIV를 이용하여 지도 객체를 생성하고 이를 앞서 정의해 둔 map 객체에 저장하고 있습니다.

다음으로 VWorld에서 제공하는 배경맵을 사용해 보도록 하겠습니다. VWorld의 2D 배경맵은 크게 2가지로 구성됩니다. 첫째는 base 맵이고 둘째는 Satellite 맵 + Hybrid 맵입니다. 두번째 맵은 2개의 TMS를 사용하여 조합됨으로써 사용자에게 제공됩니다.

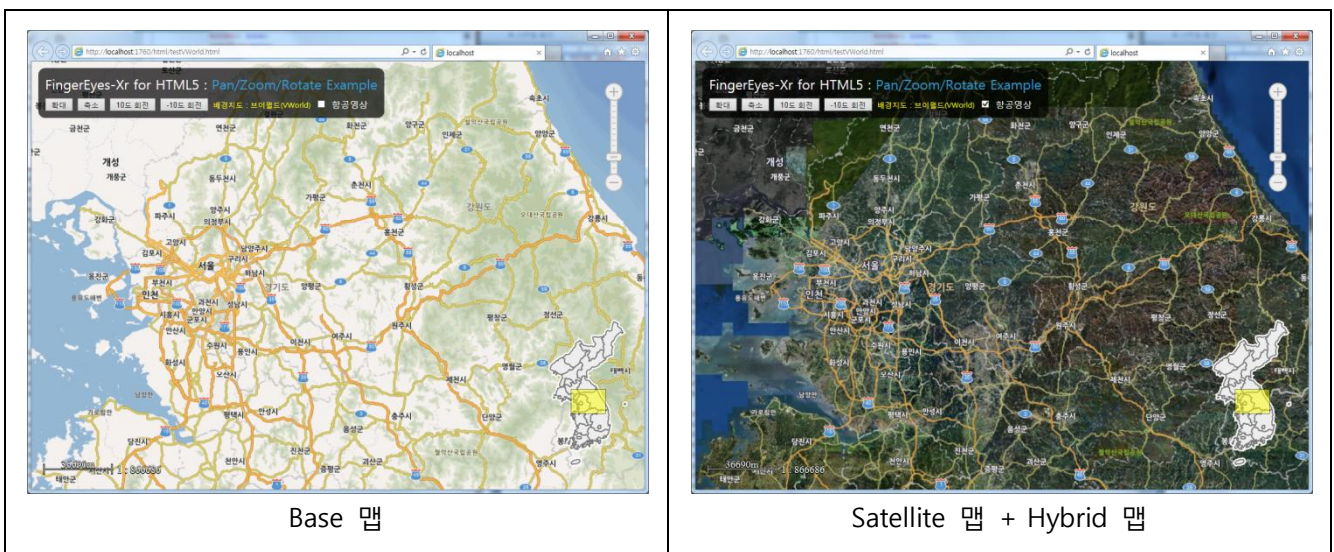


그림 4. Base 맵과 Satellite 맵 + Hybrid 맵

우리는 이 두 개의 맵을 상호 전환해 가면서 살펴볼 수 있도록 하겠습니다. 처음 지도 웹 페이지가 표시 되면 Base 맵은 표시되고 Satellite 맵 + Hybrid 맵은 표시되지 않도록 할 것입니다. 먼저 Base 맵을 위해 TMS 레이어를 생성해 lyrBase 객체에 저장해 놓습니다.

```

01 var lyrBase = new Xr.layers.TMSLayer(
02     "baseMap",
03     {
04         urls: ["http://xdworld.vworld.kr:8080/2d/Base/201310/${z}/${x}/${y}.png"],
05         upps: [
06             78000, 39000, 19600, 9800, 4900, 2400, //Dummy
07             1222.9924523925781, 611.4962261962891, 305.74811309814453,
08             152.87405654907226, 76.43702827453613, 38.218514137268066,
09             19.109257068634033, 9.554628534317017, 4.777314267158508,
10             2.388657133579254, 1.194328566789627, 0.5971642833948135
11         ],
12     },
13     mbr: new Xr.MBR(
14         -20037508.342789244, -20037508.342789244,
15         20037508.342789244, 20037508.342789244),
16     imageSize: 256,
17     reversedRows: true
18 );
19 
```

코드 8. Base 맵에 대한 TMS 레이어 생성

상당히 복잡한 내용이지만 이 내용은 TMS 형식의 지도에 필요한 정보입니다. 이러한 정보를 이용해 TMSLayer 객체를 생성하고 lyrBase 변수에 담고 있습니다.

TMSLayer 객체를 생성할 때 필요한 인자는 2개입니다. 첫번째 인자는 레이어의 이름으로 고유한 문자열 값으로써 이 문자열 값을 통해 레이어를 얻어와 참조할 수 있습니다. 그리고 두번째 인자는 TMS 형식의 지도를 생성할 때 필요한 정보로서 앞의 코드를 보면 urls, upps, mbr, imageSize, reversedRows와 같은 속성입니다.

다음은 Satellite 맵 + Hybrid 맵 중 Satellite 맵에 대한 TMS 레이어를 생성하는 코드입니다.

```

01 var lyrSatellite = new Xr.layers.TMSLayer(
02     "satelliteMap",
03     {
04         urls:
05         ["http://xdworld.vworld.kr:8080/2d/Satellite/201301/${z}/${x}/${y}.jpeg"],
06         upps: [
07             78000, 39000, 19600, 9800, 4900, 2400, //Dummy
08             1222.9924523925781, 611.4962261962891, 305.74811309814453,
09             152.87405654907226, 76.43702827453613, 38.218514137268066,
10             19.109257068634033, 9.554628534317017, 4.777314267158508,
11             2.388657133579254, 1.194328566789627, 0.5971642833948135
12         ],
13     },
14     mbr: new Xr.MBR(
15         -20037508.342789244, -20037508.342789244,
16         20037508.342789244, 20037508.342789244),
17     imageSize: 256,
18     reversedRows: true
19 );
20 
```


코드 9. Satellite 맵에 대한 TMS 레이어 생성

위의 코드에서 처럼 TMS 정보를 통해 TMS 레이어를 생성하고 이를 lyrSatellite 객체에 저장하고 있습니다.

다음은 Satellite 맵 + Hybrid 맵 중 Hybrid 맵에 대한 TMS 레이어를 생성하는 코드입니다.

```

01 var lyrHybrid = new Xr.layers.TMSLayer(
02     "hybridMap",
03     {
04         urls:
05         ["http://xdworld.vworld.kr:8080/2d/Hybrid/201310/${z}/${x}/${y}.png"],
06         upps: [
07             78000, 39000, 19600, 9800, 4900, 2400, //Dummy
08             1222.9924523925781, 611.4962261962891, 305.74811309814453,
09             152.87405654907226, 76.43702827453613, 38.218514137268066,
10             19.109257068634033, 9.554628534317017, 4.777314267158508,
11             2.388657133579254, 1.194328566789627, 0.5971642833948135 ],
12         mbr: new Xr.MBR(
13             -20037508.342789244, -20037508.342789244,
14             20037508.342789244, 20037508.342789244),
15         imageSize: 256,
16         reversedRows: true
17     }
18 );

```

코드 10. Hybrid 맵에 대한 TMS 레이어 생성

다른 TMS 레이어와 마찬가지로 TMS 정보를 통해 TMS 레이어를 생성하고 이를 lyrHybird 객체에 저장하고 있습니다.

이렇게 생성한 3개의 TMS 레이어를 앞서 DIV와 연결해 놓은 map 객체의 레이어 관리자에 추가합니다.

```

01 var lm = map.layers();
02 lm.add(lyrBase);
03 lm.add(lyrSatellite);
04 lm.add(lyrHybrid);

```

코드 11. TMS 레이어 추가

base 맵에 해당하는 것이 lyrBase 변수이고 Satellite 맵 + Hybrid 맵에 해당하는 것이 lyrSatellite 변수와 lyrHybrid 변수입니다. 이 두 개의 지도 간의 전환을 위해 displayBaseMap이라는 함수를 다음처럼 만들어 둡니다.

```

01 function diplayBaseMap(bShow) {
02     var baseMap = map.layers("baseMap");
03     var satelliteMap = map.layers("satelliteMap");
04     var hybridMap = map.layers("hybridMap");
05
06     baseMap.visibility().visible(bShow);
07     satelliteMap.visibility().visible(!bShow);
08     hybridMap.visibility().visible(!bShow);
09 }

```

코드 12. TMS 레이어 전환 함수

이 함수의 인자인 bShow를 true로 주면 base 맵에 해당하는 lyrBase의 가시화를 on 시키고 Satellite 맵 + Hybrid 맵에 해당하는 lyrSatellite와 lyrHybrid의 가시화를 off 시킵니다. bShow를 false로 주면 그 반대가 됩니다. 이 함수를 이용하면 쉽게 base 맵과 Satellite 맵 + Hybrid 맵 간의 상호 전환을 할 수 있습니다. 이 함수를 코드 11. TMS 레이어 추가에서 4번 줄 아래에 다음과 같은 코드를 추가합니다.

```
01 var lm = map.layers();
02 lm.add(lyrBase);
03 lm.add(lyrSatellite);
04 lm.add(lyrHybrid);
05
06 displayBaseMap(true);
```

코드 13. TMS 레이어 표시 여부 지정 함수 호출

displayBaseMap 호출 시에 인자를 true를 지정함으로써 base 맵은 표시되고 Satellite 맵 + Hybrid 맵은 표시하지 않고 있습니다.

3개의 TMS 레이어를 추가했고 필요한 레이어의 표시 여부까지 지정했으니 실행하면 우리가 원하는 지도가 표시될 것으로 생각할 수 있으나 아직 남은 것이 있습니다. 바로 3개의 레이어가 모두 지도 객체인 map에 완전히 추가되고 난 뒤에 화면상에 표시할 지도의 위치와 축척을 얼마로 할 것인가입니다. 이에 대한 코드는 코드 13. TMS 레이어 표시 여부 지정 함수 호출에서 6번 코드 아래에 다음과 같이 입력하면 됩니다.

```
01 map.onLayersAllReady(function() {
02     var cm = map.coordMapper();
03
04     cm.moveTo(14289580, 4436964);
05
06     cm.zoomByMapScale(3385.4915111811065);
07
08     map.update();
09 });
```

코드 14. 모든 레이어 준비 완료 이벤트에서 지도 표시 위치 및 축척 지정

코드를 좀더 상세히 설명하면, 2번은 지도 좌표를 조작하는 CoordMapper 객체를 얻어와 cm 변수에 저장하고 있습니다. CoordMapper는 지도 좌표와 화면 좌표계 간의 전환 기능과 지도의 이동, 확대, 축소 그리고 회전에 대한 기능을 제공합니다. 4번은 지도 좌표 (14289580, 4436964)를 화면 중심에 위치하도록 하는 것이고 6번은 지도 축척을 1 : 3385.4915111811065로 정하는 것입니다. 최종적으로 8번 코드에서 호출하는 map.update()를 통해 지도가 앞서 정해진 좌표와 축척값으로 화면에 그려지게 됩니다. 실행하면 다음과 같은 화면을 볼 수 있습니다.

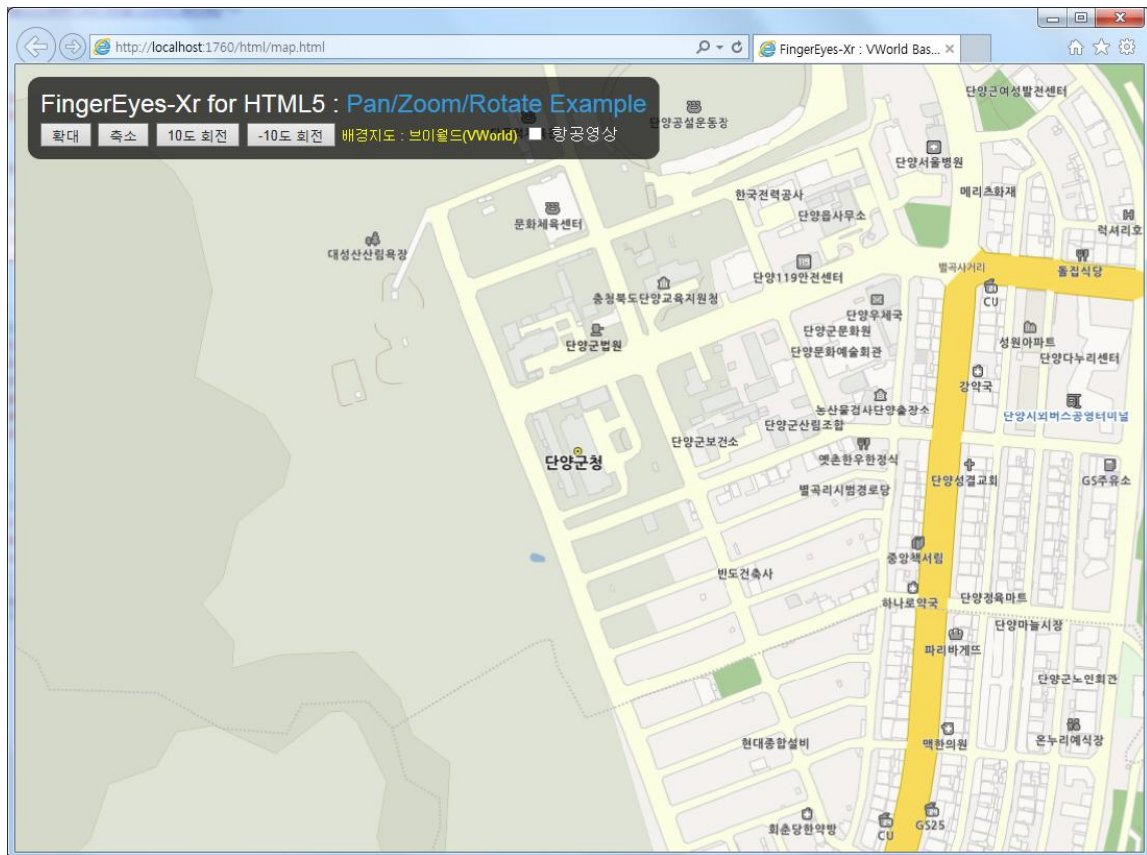


그림 5. 실행 화면

이제 4개의 Button과 하나의 CheckBox의 기능에 대한 코드를 추가 하겠습니다. 이미 이 5개의 컨트롤에는 사용자가 클릭했을 때와 체크박스의 체크 상태가 변경될 때에 대한 이벤트 함수가 지정 되어 있습니다. 그러나 아직 이 이벤트 함수를 추가하지 않았는데, 여기서 이벤트 함수를 추가 하겠습니다.

먼저 확대 버튼에 대한 클릭 이벤트 함수로 지정한 onZoomInMap 함수에 대한 코드는 다음과 같습니다.

```
01 function onZoomInMap() {
02     var cm = map.coordMapper();
03     cm.zoomByMapScale(cm.mapScale() * 0.5);
04     map.update();
05 }
```

코드 15. 지도 확대 버튼에 대한 클릭 이벤트 코드

그리고 축소 버튼에 대한 클릭 이벤트 함수로 지정한 onZoomOutMap 함수에 대한 코드는 다음과 같습니다.

```
01 function onZoomOutMap() {
02     var cm = map.coordMapper();
03     cm.zoomByMapScale(cm.mapScale() * 2);
04     map.update();
05 }
```

코드 16. 지도 축소 버튼에 대한 클릭 이벤트 코드

10도 회전 버튼에 대한 클릭 이벤트 함수로 지정한 onRotateMap 함수에 대한 코드는 다음과 같습니다.

```
01 function onRotateMap() {
02     var cm = map.coordMapper();
03     cm.rotate(10);
04     map.update();
05 }
```

코드 17. 지도 회전 버튼에 대한 클릭 이벤트 코드

-10도 회전 버튼에 대한 클릭 이벤트 함수로 지정한 onRevRotateMap 함수에 대한 코드는 다음과 같습니다.

```
01 function onRevRotateMap() {
02     var cm = map.coordMapper();
03     cm.rotate(-10);
04     map.update();
05 }
```

코드 18. 지도 반대 회전 버튼에 대한 클릭 이벤트 코드

이미 CoordMapper 객체에 대해 언급하였듯이, 이 객체를 이용하여 지도를 확대하고 축소할 수 있으며 이동 및 회전이 가능합니다.

그리고 지도를 전환하는 CheckBox에 대한 체크 변경 이벤트 함수로 지정한 onBaseMapChanged 함수는 다음과 같습니다.

```
01 function onBaseMapChanged(/* CheckBox */ cb) {
02     displayBaseMap(!cb.checked);
03     map.update();
04 }
```

코드 19. 항공영상 전환 체크 박스에 대한 이벤트 코드

앞서 만들어 놓은 displayBaseMap 함수를 이용하여 체크 여부를 인자로 호출하면 쉽게 지도 간의 전환 기능을 완성할 수 있습니다.

이제 지도를 조작할 수 있는 인덱스 맵, 줌 레벨 컨트롤, 축척바 컨트롤을 화면에 배치해 보도록 하겠습니다. 먼저 축척바 컨트롤을 화면 상에 배치하도록 하겠습니다. 아래의 코드를 load 함수의 마지막 줄에 추가합니다.

```
01 var ctrl = new Xr.ui.ScaleBarControl("sbc", map);
02 map.userControls().add(ctrl);
```

코드 20. 축척바 컨트롤 추가

축척바 컨트롤 생성을 위해 ScaleBarControl 객체를 생성합니다. 첫번째 인자는 이 컨트롤에 대한 고유한 이름입니다. 이 이름을 통해 컨트롤에 접근이 가능합니다. 실행하면 화면 좌측 하단에 축척바 컨트롤이 표시됩니다.

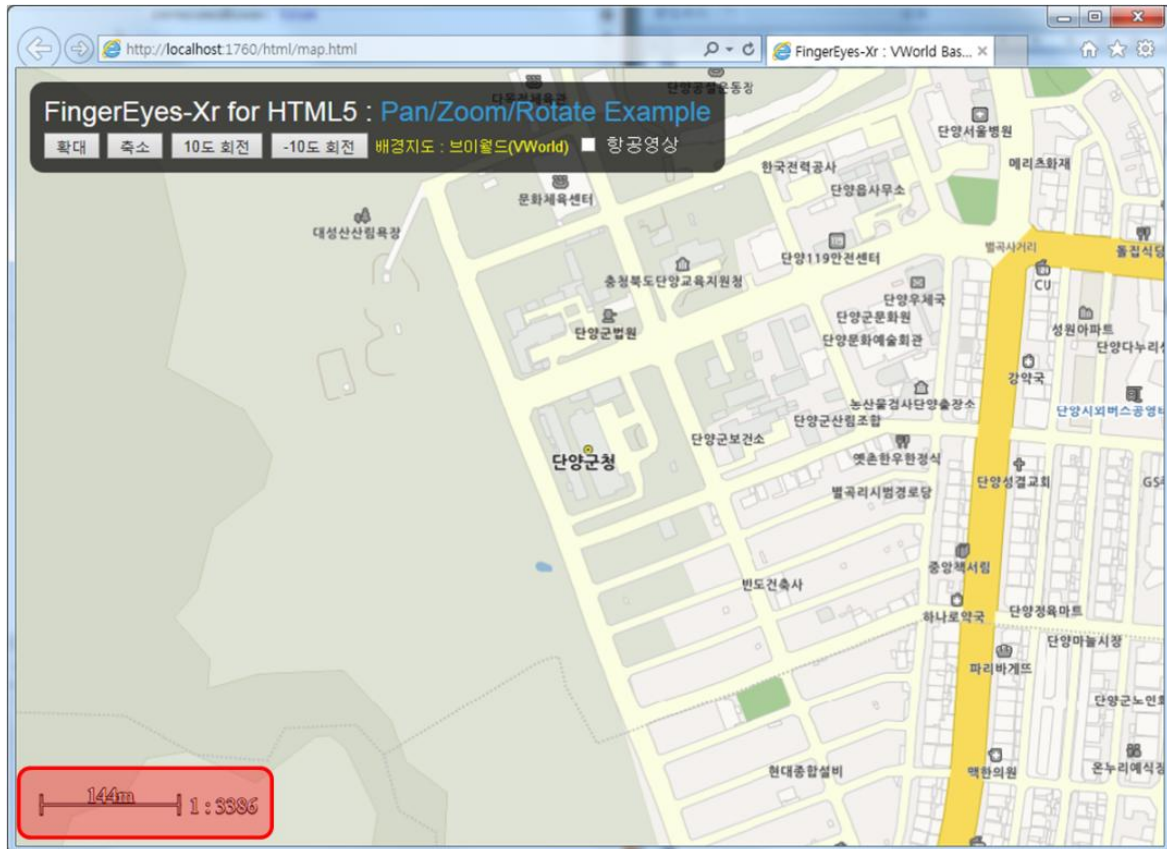


그림 6. 축척바 컨트롤 추가

다음으로 줌 레벨 컨트롤을 지도 화면에 표시하도록 하겠습니다. 이 줌 레벨 컨트롤을 표시하기 위해 load 함수의 마지막 줄에 아래의 코드를 추가합니다.

```
01 var ctrl2 = new Xr.ui.ZoomLevelControl("zlc", map);
02 ctrl2.mapScales([
    1692.7457555905532, 3385.4915111811065, 6770.983022362213,
    13541.966044724426, 27083.932086614208, 54167.864173228416,
    108335.72834645683, 216671.45669291366, 433342.9133858273,
    866685.8267716546, 1733371.6535433093, 3466743.3070866186]);
03
04 map.userControls().add(ctrl2);
```

코드 21. 줌 레벨 컨트롤 추가

위의 코드에서 2번 코드에서 마우스 휠 조작이나 컨트롤 조작을 통해 반영되는 지도의 축척에 대한 분모 값을 배열 형태로 지정합니다. 실행하면 아래의 화면처럼 줌 레벨 컨트롤이 화면에 표시되는 것을 볼 수 있습니다.

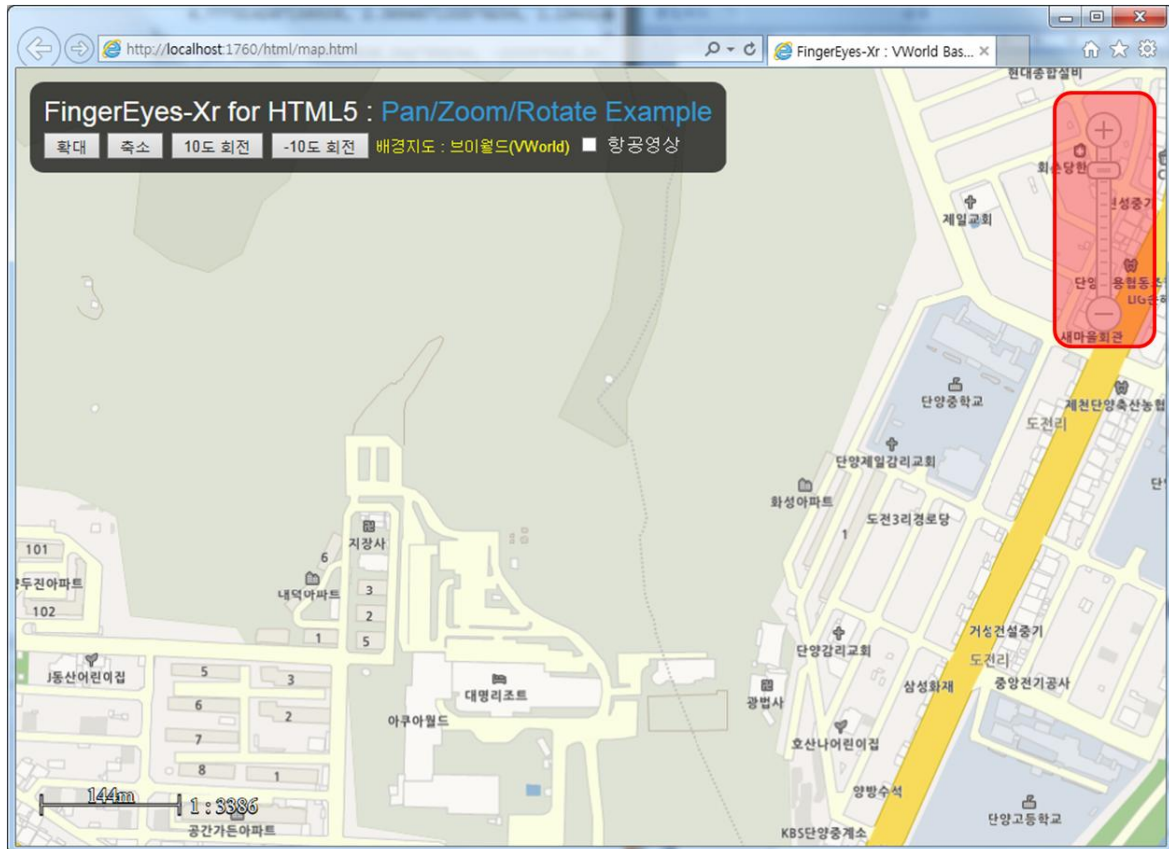


그림 7. 줌 레벨 컨트롤 추가

끝으로 인덱스 맵 컨트롤을 화면에 표시하도록 하겠습니다. 아래의 코드를 load 함수의 가장 마지막 줄에 추가합니다.

```
01 var ctrl3 = new Xr.ui.IndexMapControl("imc", map,
02     "http://222.237.78.208:8080/Xr?layerName=KOREA");
03 ctrl3.size(160, 300);
04 map.userControls().add(ctrl3);
```

코드 22. 인덱스 맵 컨트롤 추가

인덱스 맵 컨트롤 생성을 위해 IndexMapControl 객체를 생성하는데, 이때 생성자의 세번째 인자에 인덱스 맵에 표시되는 지도에 대한 Connection String을 지정하게 됩니다. 그리고 3번 코드에서 인덱스맵 컨트롤의 크기를 160px, 세로 300px로 지정하고 있습니다. 실행하면 다음처럼 인덱스 맵이 화면에 표시되는 것을 볼 수 있습니다.

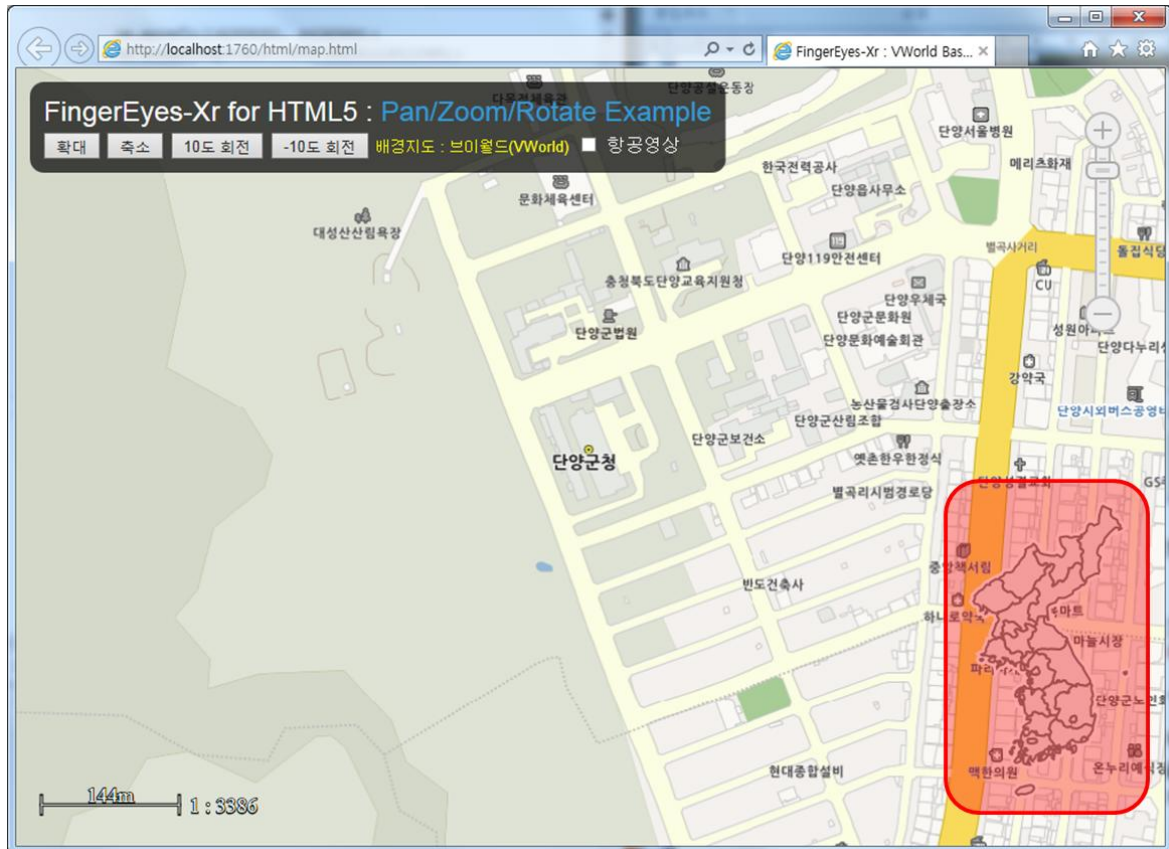


그림 8. 인덱스 맵 컨트롤 추가

이상으로 VWorld에서 제공하는 TMS 형식의 배경지도를 FingerEyes-Xr에서 사용하는 내용과 다양한 지도 조작 컨트롤을 추가하는 내용에 대해 살펴보았습니다. 끝으로 지금까지 작성한 코드에 대한 전체 내용은 아래와 같습니다.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <style>
    body
    {
      margin:0px;
      padding:0px;
    }

    #mainLayout
    {
      position:relative;
      width:100%;
      height:100%;
      border: none;
    }

    #mapDiv {
      top:0px;
      left:0px;
      position:relative;
      width:100%;
      height:100%;
      border: none;
      overflow:auto;
    }
  </style>
</head>
```

```

    }

    #title {
        top:12px;
        left:12px;
        padding: 12px;
        position:absolute;
        background:rgba(0,0,0,0.7);
        border:none;
        overflow:auto;
        border-radius: 12px;
        font-size: 24px;
        color: #ffffff;
        font-family: Arial;
    }
</style>

<title>FingerEyes-Xr : VWorld Basemap</title>

<script src="http://www.geoservice.co.kr/z/Xr.min.1.0.js"></script>

<script type="text/javascript">
    var map = null;

    function diplayBaseMap(bShow) {
        var baseMap = map.layers("baseMap");
        var satelliteMap = map.layers("satelliteMap");
        var hybridMap = map.layers("hybridMap");

        baseMap.visibility().visible(bShow);
        satelliteMap.visibility().visible(!bShow);
        hybridMap.visibility().visible(!bShow);
    }

    function load() {
        map = new Xr.Map("mapDiv", {});

        var lyrBase = new Xr.layers.TMSLayer(
            "baseMap",
            {
                urls:
["http://xdworld.vworld.kr:8080/2d/Base/201310/${z}/${x}/${y}.png"],
                upps: [
                    78000, 39000, 19600, 9800, 4900, 2400, //Dummy
                    1222.9924523925781, 611.4962261962891, 305.74811309814453,
                    152.87405654907226, 76.43702827453613, 38.218514137268066,
                    19.109257068634033, 9.554628534317017, 4.777314267158508,
                    2.388657133579254, 1.194328566789627, 0.5971642833948135
                ],
                mbr: new Xr.MBR(-20037508.342789244, -20037508.342789244,
                    20037508.342789244, 20037508.342789244),
                imageSize: 256,
                reversedRows: true
            }
        );

        var lyrSatellite = new Xr.layers.TMSLayer(
            "satelliteMap",
            {
                urls:
["http://xdworld.vworld.kr:8080/2d/Satellite/201301/${z}/${x}/${y}.jpeg"],
                upps: [
                    78000, 39000, 19600, 9800, 4900, 2400, //Dummy
                    1222.9924523925781, 611.4962261962891, 305.74811309814453,

```



```

        152.87405654907226, 76.43702827453613, 38.218514137268066,
        19.109257068634033, 9.554628534317017, 4.777314267158508,
        2.388657133579254, 1.194328566789627, 0.5971642833948135
    ],
    mbr: new Xr.MBR(-20037508.342789244, -20037508.342789244,
                    20037508.342789244, 20037508.342789244),
    imageSize: 256,
    reversedRows: true
}
);

var lyrHybrid = new Xr.layers.TMSLayer(
    "hybridMap",
    {
        urls:
["http://xdworld.vworld.kr:8080/2d/Hybrid/201310/${z}/${x}/${y}.png"],
        upps: [
            78000, 39000, 19600, 9800, 4900, 2400, //Dummy
            1222.9924523925781, 611.4962261962891, 305.74811309814453,
            152.87405654907226, 76.43702827453613, 38.218514137268066,
            19.109257068634033, 9.554628534317017, 4.777314267158508,
            2.388657133579254, 1.194328566789627, 0.5971642833948135
        ],
        mbr: new Xr.MBR(-20037508.342789244, -20037508.342789244,
                        20037508.342789244, 20037508.342789244),
        imageSize: 256,
        reversedRows: true
    }
);

var lm = map.layers();
lm.add(lyrBase);
lm.add(lyrSatellite);
lm.add(lyrHybrid);

diplayBaseMap(true);

map.onLayersAllReady(function () {
    var cm = map.coordMapper();

    cm.moveTo(14289580, 4436964);
    cm.zoomByMapScale(3385.4915111811065);

    map.update();
});

var ctrl = new Xr.ui.ScaleBarControl("sbc", map);
map.userControls().add(ctrl);

var ctrl2 = new Xr.ui.ZoomLevelControl("zlc", map);
ctrl2.mapScales([
    1692.7457555905532, 3385.4915111811065, 6770.983022362213,
    13541.966044724426, 27083.932086614208, 54167.864173228416,
    108335.72834645683, 216671.45669291366, 433342.9133858273,
    866685.8267716546, 1733371.6535433093, 3466743.3070866186]);
map.userControls().add(ctrl2);

var ctrl3 = new Xr.ui.IndexMapControl("imc", map,
    "http://222.237.78.208:8080/Xr?layerName=KOREA");
ctrl3.size(160, 300);
map.userControls().add(ctrl3);
}

function onRotateMap() {

```

```

        var cm = map.coordMapper();
        cm.rotate(10);
        map.update();
    }

    function onRevRotateMap() {
        var cm = map.coordMapper();
        cm.rotate(-10);
        map.update();
    }

    function onZoomInMap() {
        var cm = map.coordMapper();
        cm.zoomByMapScale(cm.mapScale() * 0.5);
        map.update();
    }

    function onZoomOutMap() {
        var cm = map.coordMapper();
        cm.zoomByMapScale(cm.mapScale() * 2);
        map.update();
    }

    function onBaseMapChanged(/* CheckBox */ cb) {
        diplayBaseMap(!cb.checked);
        map.update();
    }
</script>
</head>
<body onload="load()">
    <div id="mainLayout">
        <div id="mapDiv"></div>

        <div id="title">
            FingerEyes-Xr for HTML5 :
            <font color="#349bd6">Pan/Zoom/Rotate Example</font>
            <br />
            <input type="button" value="확대" onclick="onZoomInMap();" />
            <input type="button" value="축소" onclick="onZoomOutMap();" />
            <input type="button" value="10도 회전" onclick="onRotateMap();" />
            <input type="button" value="-10도 회전" onclick="onRevRotateMap();" />
            <font size="2" color="#ffff00">
            <span> 배경지도 : 브이월드 (VWorld)</span></font>
            <input type="checkbox" onchange="onBaseMapChanged(this)" />
            <font size="3" color="#ffffff"><span>항공영상</span></font>
        </div>
    </div>
</body>
</html>

```