



INTERNATIONAL STANDARD ISO/IEC 14496-2:2004 TECHNICAL CORRIGENDUM 5

Published 2013-11-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Coding of audio-visual objects — Part 2: Visual

TECHNICAL CORRIGENDUM 5

Technologies de l'information — Codage des objets audiovisuels —

Partie 2: Codage visuel

RECTIFICATIF TECHNIQUE 5

Technical Corrigendum 5 to ISO/IEC 14496-2:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

In 7.7.2.1, replace the following pseudo code:

```
field_motion_compensate_one_reference(  
    luma_pred, cb_pred, cr_pred, /* Prediction component pel array */  
    luma_ref, cb_ref, cr_ref,    /* Reference VOP pel arrays */  
    mv_top_x, mv_top_y,          /* top field motion vector */  
    mv_bot_x, mv_bot_y,          /* bottom field motion vector */  
    top_field_ref,               /* top field reference */  
    bottom_field_ref,            /* bottom field reference */  
    x, y,                       /* current luma macroblock coords */  
    rounding_type)               /* rounding type */
```

```

{
    mc(luma_pred, luma_ref, x, y, 16, 16, mv_top_x, mv_top_y,
        rounding_type, 0, top_field_ref, 2, "y");
    mc(luma_pred, luma_ref, x, y, 16, 16, mv_bot_x, mv_bot_y,
        rounding_type, 1, bottom_field_ref, 2, "y");
    mc(cb_pred, cb_ref, x/2, y/2, 8, 8,
        Div2Round(mv_top_x), Div2Round(mv_top_y/2)*2,
        rounding_type, 0, top_field_ref, 2, "u");
    mc(cr_pred, cr_ref, x/2, y/2, 8, 8,
        Div2Round(mv_top_x), Div2Round(mv_top_y/2)*2,
        rounding_type, 0, top_field_ref, 2, "u");
    mc(cb_pred, cb_ref, x/2, y/2, 8, 8,
        Div2Round(mv_bot_x), Div2Round(mv_bot_y/2)*2,
        rounding_type, 1, bottom_field_ref, 2, "v");
    mc(cr_pred, cr_ref, x/2, y/2, 8, 8,
        Div2Round(mv_bot_x), Div2Round(mv_bot_y/2)*2,
        rounding_type, 1, bottom_field_ref, 2, "v");
}

```

with:

```

field_motion_compensate_one_reference(
    luma_pred, cb_pred, cr_pred, /* Prediction component pel array */
    luma_ref, cb_ref, cr_ref,    /* Reference VOP pel arrays */
    mv_top_x, mv_top_y,          /* top field motion vector */
    mv_bot_x, mv_bot_y,          /* bottom field motion vector */
    top_field_ref,               /* top field reference */
    bottom_field_ref,            /* bottom field reference */
    x, y,                        /* current luma macroblock coords */
    rounding_type)               /* rounding type */
{
    mc(luma_pred, luma_ref, x, y, 16, 16, mv_top_x, mv_top_y,
        rounding_type, 0, top_field_ref, 2, "y");
    mc(luma_pred, luma_ref, x, y, 16, 16, mv_bot_x, mv_bot_y,
        rounding_type, 1, bottom_field_ref, 2, "y");
    mc(cb_pred, cb_ref, x/2, y/2, 8, 8,
        Div2Round(mv_top_x), Div2Round(mv_top_y>>1)*2,
        rounding_type, 0, top_field_ref, 2, "u");
    mc(cr_pred, cr_ref, x/2, y/2, 8, 8,
        Div2Round(mv_top_x), Div2Round(mv_top_y>>1)*2,
        rounding_type, 0, top_field_ref, 2, "u");
    mc(cb_pred, cb_ref, x/2, y/2, 8, 8,
        Div2Round(mv_bot_x), Div2Round(mv_bot_y>>1)*2,
        rounding_type, 1, bottom_field_ref, 2, "v");
    mc(cr_pred, cr_ref, x/2, y/2, 8, 8,
        Div2Round(mv_bot_x), Div2Round(mv_bot_y>>1)*2,
        rounding_type, 1, bottom_field_ref, 2, "v");
}

```