Subject: Alpha composition OVER operator WITHOUT premultiplied alpha
Posted by mirek on Thu, 27 Jul 2006 14:42:05 GMT
View Forum Message <> Reply to Message

...well, I was not able to google the effective code for this one, so I had to invent it myself :

```
struct sBlends {
 int16 m;
 byte  a;
};

sBlends *sblends;

void sOnceInitBlends()
{
 ONCELOCK {
  RTIMING("InitBlends");
  sblends = (sBlends *)MemoryAllocPermanent(256 * 256 * sizeof(sBlends));
  for(int Fa = 0; Fa <= 255; Fa++)
   for(int Ba = 0; Ba <= 255; Ba++) {
    double A = (Fa / 255.0 + Ba / 255.0 - Fa / 255.0 * Ba / 255.0);
    double dDa = 255 * A;
    sblends[(Ba << 8) + Fa].a = minmax((int)(255 * A + 0.5), 0, 255);
    sblends[(Ba << 8) + Fa].m = A > 0.001 ? int(256 * (Fa / 255.0) / A + 0.5) : 0;
   }
 }
}

inline void sInitBlends()
{
 if(!sblends)
  sOnceInitBlends();
}

void AlphaBlend(RGBA *b, const RGBA *f, int len)
{
 RTIMING("Full");
 sInitBlends();
 const RGBA *e = f + len;
 while(f < e) {
  sBlends& x = sblends[(b->a << 8) + f->a];
  int m = x.m;
  b->a = x.a;
  b->r += m * (f->r - b->r) >> 8;
  b->g += m * (f->g - b->g) >> 8;
  b->b += m * (f->b - b->b) >> 8;
  b++;
```

```
  f++;
 }
}
```

..my measurements show it only 20% slower on CPU with 128KB cache than premultiplied alpha code.

Mirek
