

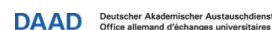
MÉDIAS OPTIQUES COURS BERLINOIS 1999

L'engouement dont la théorie des médias initiée par Friedrich Kittler († 2011) a fait l'objet dans les pays de langue allemande n'a eu en France que peu d'écho, en raison notamment de l'absence quasi totale de traduction. Dispensés sous forme de cours à l'Université Humboldt de Berlin, où Kittler a occupé jusqu'en 2009 la chaire d'Esthétique et d'histoire des médias, les *Médias optiques* constituent une porte d'entrée particulièrement claire et vivante sur les *Medienwissenschaften*. Le projet kittlérien d'expulser l'esprit des sciences humaines est ici appliqué au domaine des images, dont il s'agit de dégager l'a priori média-technique. Il en résulte une relecture historique et systématique du domaine du visible qui analyse de manière originale et parfois provocante le passage des beaux-arts (peinture, littérature) aux médias optiques (photographie, cinéma, télévision) puis au médium de tous les médias qu'est l'ordinateur.

Traduit sous la direction d'Audrey Rieber, maître de conférences en philosophie (ens de Lyon) par une équipe réunissant : Anaïs Carvalho, doctorante en histoire de l'art (Montpellier 3), Tamara Eble, normalienne, agrégée d'allemand, doctorante en études germaniques (ens de Lyon), Ève Vayssière, certifiée d'allemand, docteur en études germaniques (Paris 3), Slaven Waelti, docteur en Littérature française, assistant à l'Université de Bâle.

Illustration de couverture : © Peter Berz, Susanne Holl

Ouvrage publié avec le soutien de :



Maison des Sciences de l'Homme
Paris Nord

DEUTSCHES FORUM FÜR KUNSTGESCHICHTE
CENTRE ALLEMAND D'HISTOIRE DE L'ART



MÉDIAS OPTIQUES
COURS BERLINOIS 1999
Friedrich Kittler
esthétiques → allemandes

```
case torus;
tp = Obj->UU.torptr;
Obj->umkugel.m = tp->m;
Obj->umkugel.rad = tp->a + tp->b;
tp->a *= tp->a;
tp->b *= tp->b;
break;
case ellipsoid;
ep = Obj->UU.ellptr;
ep->mat.a = r / (ep->ax.x * ep->ax.x);
ep->mat.e = s / (ep->ax.y * ep->ax.y);
ep->mat.h = t / (ep->ax.z * ep->ax.z); // Hyperboloid = -h
ep->mat.d = -ep->m.x*r;
ep->mat.g = -ep->m.y*s;
ep->mat.i = -ep->m.z*t; // Hyperboloid = -i
ep->mat.j = (ep->m.x*ep->m.x*r + ep->m.y*ep->m.y*s
+ ep->m.z*ep->m.z*t) - 1;
Obj->umkugel.m = ep->m;
r = fabs(ep->ax.x);
s = fabs(ep->ax.y);
Obj->umkugel.rad = t = fabs(ep->ax.z);
if (r > s)
{
if (r > t)
Obj->umkugel.rad = r;
}
else
{
if (s > t)
Obj->umkugel.rad = s;
}
break;
case kegel;
cp = Obj->UU.cone;
r = cp->apex.rad * cp->apex.rad;
Obj->umkugel.m = cp->apex.m;
Obj->umkugel.m.z = halb*(r+1) * (cp->high-cp->low) - r * cp->apex.m.z;
Obj->umkugel.rad = r * (cp->apex.m.z-cp->high) * (cp->apex.m.z-cp->high)
+ (cp->high-Obj->umkugel.m.z) * (cp->apex.m.z-cp->high);
Obj->umkugel.rad = sqrt(r);
break;
case pyramide;
py = Obj->UU.pyrptr;
py->planes = 4;
for (i = 0; i < 4; i++)
{
py->flaechen[i].vertnum = 3;
py->flaechen[i].polynom = 4;
}
ganz = py->flaechen[0].vertnum * py->planes;
Obj->umkugel.rad = 0;
Obj->umkugel.m.x = Obj->umkugel.m.y = Obj->umkugel.m.z = 0;
for (i = 0; i < py->planes; i++)
for (j = 0; j < py->flaechen[i].vertnum; j++)
vaddq(&Obj->umkugel.m, &py->flaechen[i].vtx[j]);
vscaleq(&Obj->umkugel.m, 1 / (float)ganz);
for (i = 0; i < py->planes; i++)
for (j = 0; j < py->flaechen[i].vertnum; j++)
if (r = vdst(&Obj->umkugel.m, &py->flaechen[i].vtx[j])
> Obj->umkugel.rad)
Obj->umkugel.rad = r;
for (i = 0; i < py->planes; i++)
{
vsub(d, &py->flaechen[i].vtx[i], &py->flaechen[i].vtx[0]);
vsub(&d[1], &py->flaechen[i].vtx[2], &py->flaechen[i].vtx[1]);
vcross(&py->flaechen[i].normal, d, &d[1]);
normalize(&py->flaechen[i].normal);
vsubnorm(d, &py->flaechen[i].vtx[0], &Obj->umkugel.m);
// gegen falsche Reihenfolge von Polygonecken-Eingaben
negnorm(d, &py->flaechen[i].normal);
}
break;
case box;
bp = Obj->UU.boxptr;
bp->planes = 6;
for (i = 0; i < 6; i++)
{
bp->flaechen[i].vertnum = 4;
bp->flaechen[i].polynom = 6;
}
ganz = bp->flaechen[0].vertnum * bp->planes;
Obj->umkugel.rad = 0;
Obj->umkugel.m.x = Obj->umkugel.m.y = Obj->umkugel.m.z = 0;
for (i = 0; i < bp->planes; i++)
for (j = 0; j < bp->flaechen[i].vertnum; j++)
vaddq(&Obj->umkugel.m, &bp->flaechen[i].vtx[j]);
vscaleq(&Obj->umkugel.m, 1 / (float)ganz);
for (i = 0; i < bp->planes; i++)
for (j = 0; j < bp->flaechen[i].vertnum; j++)
if (r = vdst(&Obj->umkugel.m, &bp->flaechen[i].vtx[j])
> Obj->umkugel.rad)
Obj->umkugel.rad = r;
for (i = 0; i < bp->planes; i++)
{
vsub(d, &bp->flaechen[i].vtx[i], &bp->flaechen[i].vtx[0]);
vsub(&d[1], &bp->flaechen[i].vtx[2], &bp->flaechen[i].vtx[1]);
vcross(&bp->flaechen[i].normal, d, &d[1]);
normalize(&bp->flaechen[i].normal);
}
```

Friedrich Kittler

MÉDIAS OPTIQUES COURS BERLINOIS 1999

Traduit sous la direction d'Audrey Rieber
Avec une introduction de Peter Berz