This is the notebook for two rotation figures. One is for a rotation of a vector lying in a plane (but that plane is viewed from a 3D vantage point), and the other is for a rotation with respect to a plane through an angle, and applied to a vector out of the plane.

```
<< peeters` ;
peeters`setGitDir[ "../project/figures/GAelectrodynamics" ]
```

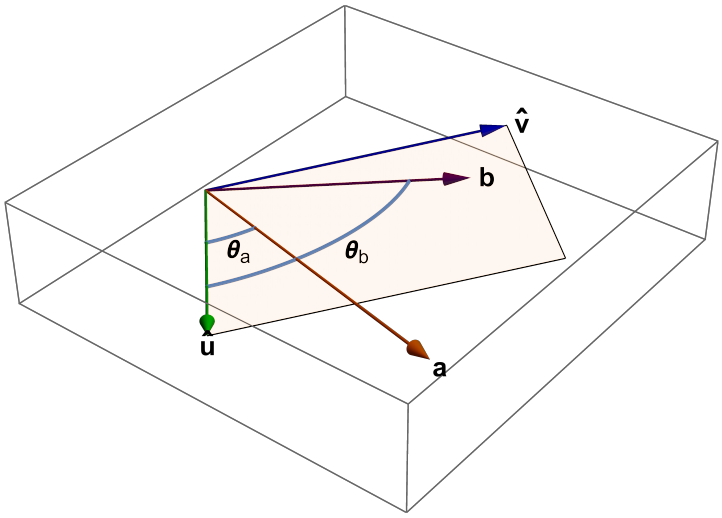
```
ClearAll[o, e1, e2, bold, sz, fs, tsub, midpoint, midtext]
o = {0, 0, 0};
e2 = {0, 1, 0};
e3 = {0, 0, 1};
e1 = {1, 0, 0};
bold = Style[ #, Bold] &;
sz = 14;
fs = Style[ #, FontSize → sz] &;
tsub[t_, s_] := Subscript[bold[t] // fs, s];
esub := tsub[e, #] &;
(*vsub := tsub[v, #]&;
midpoint[p_] := (p[[1]] + p[[2]])/2 ;
midtext[p_, sh_,text_] := Text[text,midpoint[p] + sh]*)
```

```
ClearAll[u, ucap, v, vcap, parallelogram, a, b, arc, p, polar]
parallelogram[p_, {v1_, v2_}] := Polygon[{p + v1, p + v1 + v2, p + v2, p}]

v = e1 + e2 + e3 / 4;
vcap = (v) // Normalize;
u = (e3 / 4 - e2 + e1);
ucap = ((u - u.vcap vcap) // Normalize);
polar[m_, a_] := m (ucap Cos[a] + vcap Sin[a]);
a = polar[1.5, Pi / 8];
b = polar[0.9, 3 Pi / 8];

arc[r_, s_, f_, n_ : o] :=
  ParametricPlot3D[ n + r (ucap Cos[x] + vcap Sin[x]), {x, s, f} ];

p = Show[
  {Graphics3D[ {
     Opacity[.1],
     parallelogram[o, {ucap, vcap }],
     Opacity[1],
     Green // Darker,
     Arrow[Tube[{o, ucap}]],
     Blue // Darker,
     Arrow[Tube[{o, vcap}]],
     Black,
     Text[ bold[OverHat["u"]] // fs, 1.05 ucap],
     Text[ bold[OverHat["v"]] // fs, 1.05 vcap],
     Orange // Darker,
     Arrow[Tube[{o, a}]],
     Purple // Darker,
     Arrow[Tube[{o, b}]],
     Black,
     Text[ bold["a"] // fs, a + 0.05 (a // Normalize)],
     Text[ bold["b"] // fs, b  + 0.05 (b // Normalize)],
     Text[ tsub["θ", "a"] // fs, polar[0.5, Pi / 16]],
     Text[ tsub["θ", "b"] // fs, polar[0.8, 3 Pi / 16]]
    }
   ],
   arc[0.4, 0, Pi / 8],
   arc[0.7, 0, 3 Pi / 8]
  }]
```




```
peeters`exportForLatex["ParallelogramFig1", p]
```

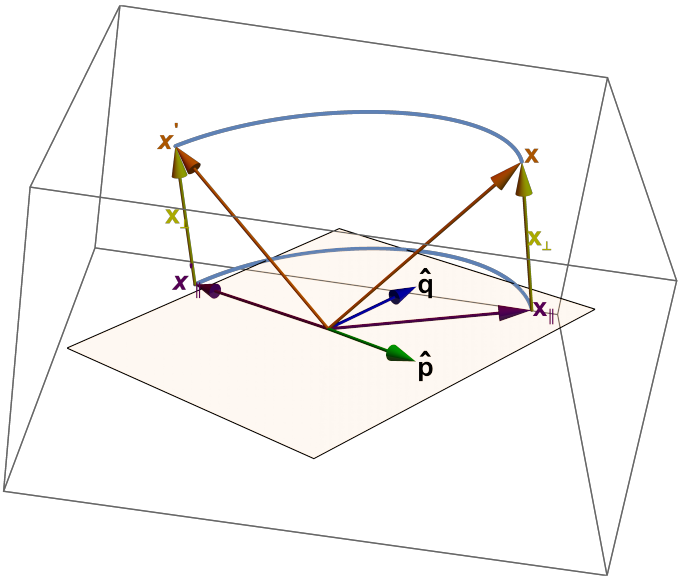
```
{ParallelogramFig1.eps, ParallelogramFig1pn.png}
```

```
ClearAll[p2, arc2, rot];
rot[xu_, xv_, xp_, uc_, vc_, nc_, th_] :=
  xu (uc Cos[th] + vc Sin[th]) + xv (vc Cos[th] - uc Sin[th]) + xp nc;
arc2[ x_, uc_, vc_, nc_, theta_] := Module[{xu, xv, xp},
   xu = x.uc;
   xv = x.vc;
   xp = x.nc;
   ParametricPlot3D[ rot[xu, xv, xp, uc, vc, nc, t], {t, 0, theta} ]
  ];

p2 = Module[{x, n, ncap, xproj, xrej, theta, xPrime, xparPrime, scale},
  n = e3 - e3.ucap ucap - e3.vcap vcap;
  ncap = n // Normalize;
  xproj = 1.1 ucap + 1.2 vcap ;
  xrej = 1.3 n;
  x = xproj + xrej;
  theta = 3 Pi / 4;
  xPrime = rot[x.ucap, x.vcap, x.ncap, ucap, vcap, ncap, theta];
  xparPrime = rot[x.ucap, x.vcap, 0, ucap, vcap, ncap, theta];
  scale = 1.5;

  Show[
   {Graphics3D[ {
      Opacity[.1],
      parallelogram[o - scale (ucap + vcap), 2 scale {ucap, vcap }],
      Opacity[1],
      Green // Darker,
```

```
      Arrow[Tube[{o, ucap}]],
      Blue // Darker,
      Arrow[Tube[{o, vcap}]],
      Black,
      Text[ bold[OverHat["p"]] // fs, 1.1 ucap],
      Text[ bold[OverHat["q"]] // fs, 1.1 vcap],
      Orange // Darker,
      Arrow[Tube[{o, x}]],
      Text[ bold["x"] // fs, x + 0.1 (x // Normalize)],
      Arrow[Tube[{o, xPrime}]],
      Text[ bold["x'"] // fs, xPrime + 0.1 (xPrime // Normalize)],
      Purple // Darker,
      Arrow[Tube[{o, xproj}]],
      Text[ tsub["x", "∥"] // fs, xproj + 0.1 (xproj // Normalize)],
      Arrow[Tube[{o, xparPrime}]],
      Text[ tsub["x'", "∥"] // fs, xparPrime + 0.1 (xparPrime // Normalize)],
      Yellow // Darker,
      Arrow[Tube[{xproj, xrej + xproj}]],
      Text[ tsub["x", "⊥"] // fs, xproj + xrej / 2 + 0.1 (xproj // Normalize)],
      Arrow[Tube[{xparPrime, xrej + xparPrime}]],
      Text[ tsub["x", "⊥"] // fs,
       xparPrime + xrej / 2 + 0.1 (xparPrime // Normalize)]
     }
    ]
    ,
    arc2[x, ucap, vcap, ncap, theta]
    ,
    arc2[xproj, ucap, vcap, ncap, theta]
   }]
]
```




```
peeters`exportForLatex["RotationFig1", p2]
```

{RotationFig1.eps, RotationFig1pn.png}