The matrix package

Here are the routines defined in the file **`matrix.inc`**. Roughly speaking, the entire package is concerned with matrix and vector manipulation. Arguments are put onto the stack before the routine is called, and return values are left on the stack at exit. Side effects are described, too.

Vectors in this scheme are simple arrays of numbers. Matrices are arrays of vectors, which are interpreted as the rows of the matrix.

ROUTINE: **vectortranslate**
ARGUMENTS: Two vectors u and v of any length
RETURNS: The vector sum $u + v$

The vectors u and v must have the same length.

ROUTINE: **dotproduct**
ARGUMENTS: Two vectors u and v of any length
RETURNS: The dot product $u \bullet v$

The vectors u and v must have the same length.

ROUTINE: **matrixvector**
ARGUMENTS: A matrix M and a vector v, of any compatible sizes
RETURNS: The product Mv

If M is $m \times n$ then v must have length n.

ROUTINE: **identitymatrix**
ARGUMENTS: An integer n
RETURNS: The $n \times n$ identity matrix I

ROUTINE: **affinemap**
ARGUMENTS: $[M\ u]\ v$ where M is an $n \times n$ matrix, u and v of dimension n
RETURNS: $Mv + u$

ROUTINE: **transpose**
ARGUMENTS: A matrix A
RETURNS: ${}^tA$

ROUTINE: **matrixmul**
ARGUMENTS: Two matrices A, B
RETURNS: AB

ROUTINE: **vectorlength**
ARGUMENTS: A vector v
RETURNS: $\|v\|$

ROUTINE: **normalized**
ARGUMENTS: A vector v
RETURNS: $v/\|v\|$

ROUTINE: **crossproduct**
ARGUMENTS: Two 3-vectors u and v
RETURNS: $u \times v$

ROUTINE: **3rotate**
ARGUMENTS: A 3-vector α, an angle θ, and a 3-vector v
RETURNS: rotation of v by θ around α

ROUTINE: **rotation-matrix**
ARGUMENTS: A 3-vector α and an angle θ
RETURNS: The matrix of the rotation by θ around α

ROUTINE: **vectorscale**
ARGUMENTS: A vector v and a scalar c
RETURNS: cv

ROUTINE: **affineconcat**
ARGUMENTS: Two affine transformations $[M_1 \ v_1]$ and $[M_2 \ v_2]$
RETURNS: The composition of these, in the same format