1 Conclusion

The software we have developed so far has room for much improvement. Firstly, the hyperstreamline module does not handle closed-hyperstreamlines. This is an important case to handle, as to prevent infinite cycling during hyperstreamline calculation [**?**].Topological features such as degenerate points are also important cases that must be handeled. The topology of a tensor field is the topology of its eigenvectors. In the case of two or more eigenvalues being equal, at least one eigenvector is linearly dependent. As a consequence, hyperstreamlines at such points, can branch out through multiple paths. Finally, perhaps the most important physical characteristic of any tensor field is its time dependency. Handling an unsteady tensor field is important as many physical processes are unsteady.

One of the main challenges of tensor field visualization is the difficulty there lies in adapting same techniques across datasets with different physical attributes. Examples of such fields include stress and strain tensors, rate of deformation tensor, and the diffusion tensor. The physical meaning of tensors can greatly impact how they should be visualized, even when the mathematical representations of these tensors are the same (as we have shown in **??**) [**?**]. Using differntial geometry, we have demonstrated a new method, where we solve the geodesic differential equations and apply similar techniques as hyperstreamlines. The method itself was applied on metric tensor. It can easily be extended to any other second order tensor. Though, there are limitation, such as the Christoffel symbol of second kind exists only if the metric is non-singular. We showed that the momentum flux density is one such tensor which can not be applied.

We can take the geodesics one step further, if we can manage to combine hyperstreamlines techniques using geodesics to determine the pricipal direction. This hybrid geodesic-hyperstreamline method requires further investigation.

The important finding of this thesis are as following : There is a gaping disparity for readily available free software which permit the user multiple visualization methods for second order (or higher order) tensor fields. As such, we set ourselves upon the daunting task of creating our own tensor module (even though we limited ourselves to a few methods). However, the process it self was quite revelatory. To create a module from almost scratch is an exiciting task, but still quite difficult. As such, much of the focus has been in the implementation process itself.